

TRƯỜNG ĐẠI HỌC TRÀ VINH  
KHOA KỸ THUẬT VÀ CÔNG NGHỆ  
BỘ MÔN CÔNG NGHỆ THÔNG TIN



**TÀI LIỆU GIẢNG DẠY  
MÔN THIẾT KẾ VÀ LẬP TRÌNH WEB**

*GV biên soạn: Đoàn Phước Miền  
Phạm Thị Trúc Mai*

Trà Vinh, ngày tháng năm 2014  
**Lưu hành nội bộ**

## MỤC LỤC

CHƯƠNG 1 .....	1
NGÔN NGỮ HTML .....	1
BÀI 1 .....	1
TỔNG QUAN VỀ INTERNET, WEB VÀ THIẾT KẾ WEB .....	1
1.1.1 Lịch sử phát triển của Internet và Web .....	1
1.1.2 Các khái niệm cơ bản trên Internet và Web .....	2
1.1.3 Các dịch vụ cơ bản trên Internet .....	4
1.1.4 Cách hoạt động của một trang Web .....	5
1.1.5 Cấu trúc trang HTML .....	6
BÀI 2 .....	9
CÁC THẺ HTML CƠ BẢN .....	9
1.2.1 Thẻ <body>, chú thích và các ký hiệu đặc biệt .....	9
1.2.2 Thẻ định dạng văn bản .....	10
1.2.2.1 Thẻ <div> ... </div> .....	10
1.2.2.2 Thẻ <p> ... </p> .....	11
1.2.2.3 Các thẻ định dạng tiêu đề .....	12
1.2.2.4 Thẻ định dạng chữ .....	13
1.2.2.5 Thẻ <marquee> .....	16
1.2.3 Thẻ định dạng danh sách .....	17
1.2.3.1 Danh sách không thứ tự <ul> .....	18
1.2.3.2 Danh sách có thứ tự .....	19
1.2.4 Thẻ chèn hình ảnh, âm thanh .....	21
BÀI 3 .....	28
SIÊU LIÊN KẾT VÀ KHUNG .....	28
1.3.1 Siêu liên kết – Hyperlink .....	28
1.3.1.1 Liên kết trong .....	29
1.3.1.2 Liên kết ngoài .....	31
1.3.2 Khung – frame .....	33
1.3.2.1 Thẻ <frameset> ... </frameset> .....	33
1.3.2.2 Thẻ <noframe> ... </noframe> .....	39
1.3.2.3 Thẻ <iframe> ... </iframe> .....	39
BÀI 4 .....	46
BẢNG BIỂU TRÊN WEB .....	46
1.4.1 Tạo bảng .....	46
1.4.2 Các thuộc tính định dạng bảng .....	46
1.4.2.1 Thuộc tính trong thẻ bảng <table> .....	46
1.4.2.2 Thuộc tính trong thẻ dòng <tr> .....	47
1.4.2.3 Thuộc tính trong các thẻ ô / cột <th>, <td> .....	48
1.4.3 Thiết kế giao diện Web bằng <table> .....	50
BÀI 5 .....	60
FORM VÀ CÁC CONTROLS .....	60
1.5.1 Ý nghĩa của form trong trang Web và cách khởi tạo .....	60
1.5.2 Các thành phần trong form .....	61
1.5.2.1 Thẻ <input> .....	61
1.5.2.2 Thẻ <textarea> ... </textarea> .....	64
1.5.2.3 Thẻ <select> ... </select> .....	64
1.5.2.4 Thẻ <fieldset> ... </fieldset> .....	65

<i>1.5.2.5 Bảng tổng hợp các thành phần trong form</i> .....	70
<b>CHƯƠNG 2 .....</b>	<b>79</b>
<b>BẢNG MẪU NẠP CHỒNG – CSS .....</b>	<b>79</b>
2.1 Khái niệm.....	79
2.2 Phân loại .....	80
2.2.1 <i>Inline style sheet</i> .....	81
2.2.2 <i>Internal style sheet</i> .....	82
2.2.3 <i>External style sheet</i> .....	84
2.3 Vận dụng CSS vào thiết kế giao diện .....	86
2.3.1 <i>Bộ chọn - Selector</i> .....	86
2.3.2 <i>Các thuộc tính CSS thông dụng trong thiết kế giao diện.</i> .....	87
<b>CHƯƠNG 3 .....</b>	<b>102</b>
<b>NGÔN NGỮ JAVASCRIPT .....</b>	<b>102</b>
3.1 Giới thiệu .....	102
3.2 Biến, toán tử, toán hạng, hằng, hàm, lệnh trong JavaScript .....	103
3.2.1 <i>Biến trong JavaScript</i> .....	103
3.2.2 <i>Toán tử</i> .....	105
3.2.3 <i>Hằng số</i> .....	106
3.2.4 <i>Hàm</i> .....	106
3.2.5 <i>Các lệnh điều khiển trong JavaScript</i> .....	109
3.2.6 <i>Tham chiếu đến các đối tượng trong HTML</i> .....	115
<b>CHƯƠNG 4 .....</b>	<b>129</b>
<b>TỔNG QUAN VỀ PHP .....</b>	<b>129</b>
BÀI 1 .....	129
GIỚI THIỆU VỀ PHP .....	129
4.1.1. PHP là gì? .....	129
4.1.2. Lịch sử phát triển PHP .....	129
4.1.3. Ưu điểm và nhược điểm của PHP .....	130
4.1.3.1. <i>Ưu điểm</i> .....	130
4.1.3.2. <i>Nhược điểm:</i> .....	131
4.1.4. Cơ chế hoạt động của Web động .....	131
BÀI 2 .....	133
GIỚI THIỆU VÀ CÀI ĐẶT WEB SERVER .....	133
4.2.1. Giới thiệu WebServer .....	133
4.2.2. Cài đặt WampServer2.....	134
4.2.3. Sử dụng WampServer 2 .....	141
<b>CHƯƠNG 5 .....</b>	<b>144</b>
<b>PHP CƠ BẢN .....</b>	<b>144</b>
BÀI 1 .....	144
CÁC QUY ƯỚC VÀ CÁCH KHAI BÁO .....	144
5.1.1. Quy ước trong PHP .....	144
5.1.2. Cách khai báo biến .....	145
5.1.3. Kiểu dữ liệu trong PHP.....	146
5.1.4. Kiểu chuỗi – string .....	146
5.1.5. Các phép toán trong PHP.....	150
5.1.5.1. <i>Các phép toán số học</i> .....	150
5.1.5.2. <i>Phép gán</i> .....	151
5.1.5.3. <i>Các phép toán so sánh</i> .....	151
5.1.5.4. <i>Phép toán logic</i> .....	152

BÀI 2.....	153
MỆNH ĐỀ ĐIỀU KIỆN VÀ CẤU TRÚC LẶP.....	153
5.2.1.    Mệnh đề điều kiện .....	153
5.2.1.1.    Mệnh đề if .....	153
5.2.1.2.    Mệnh đề if... else .....	153
5.2.1.3.    Mệnh đề if...elseif...else.....	154
5.2.1.4.    Mệnh đề switch .....	154
5.2.2.    Cấu trúc lặp .....	154
5.2.2.1.    Cấu trúc while .....	154
5.2.2.2.    Cấu trúc do...whi le.....	155
5.2.2.3.    Cấu trúc for.....	156
5.2.2.4.    Cấu trúc foreach.....	157
BÀI 3.....	159
MẢNG VÀ HÀM TRONG PHP.....	159
5.3.1.    Mảng trong PHP.....	159
5.3.1.1.    Mảng sử dụng các chỉ số .....	159
5.3.1.2.    Mảng sử dụng các chuỗi làm khóa Ví dụ: Mảng \$nation.....	159
5.3.2.    Các mảng siêu toàn cục .....	160
5.3.3.    Tạo mảng: Có nhiều cách để tạo mảng: .....	160
5.3.3.1.    Dùng hàm array().....	160
5.3.3.2.    Tạo mảng số với hàm range() .....	161
5.3.3.3.    Tạo mảng với dữ liệu bất kỳ với vòng lặp.....	161
5.3.4.    Truy xuất mảng .....	161
5.3.5.    Các hàm thông dụng xử lý mảng .....	162
5.3.6.    Mảng đa chiều .....	162
5.3.7.    Chuyển đổi giữa mảng và chuỗi.....	163
5.3.8.    Sắp xếp mảng .....	163
5.3.9.    Hàm người dùng định nghĩa .....	165
5.3.9.1.    Cú pháp:.....	165
5.3.9.2.    Sử dụng hàm .....	165
5.3.9.3.    Tầm vực của biến trong trang PHP .....	166
5.3.9.4.    Tham trị và tham biến .....	166
5.3.10.    Sử dụng lại mã PHP .....	167
5.3.11.    File trong PHP.....	169
5.3.11.1.    Mở file.....	170
5.3.11.2.    Đóng file .....	171
5.3.11.3.    Đọc và ghi file trong PHP .....	171
5.3.11.4.    Kiểm tra kết thúc file .....	172
5.3.12.    Lớp đối tượng – Class .....	175
5.3.12.1.    Cách khai báo .....	175
5.3.12.2.    Cách sử dụng.....	175
5.3.13.    Một số hàm thông dụng trong PHP.....	175
BÀI 4:.....	182
XỬ LÝ GIÁ TRỊ FORM TRONG PHP .....	182
5.4.1.    Phương thức GET.....	183
5.4.2.    Phương thức POST.....	183
BÀI 5:.....	185
KHÁI NIỆM CƠ BẢN VỀ COOKIE VÀ SESSION TRONG PHP.....	185
5.5.1.    Tổng quan về cookie .....	185

5.5.1.1.	<i>Thiết lập cookie</i>	186
5.5.1.2.	<i>Sử dụng cookie</i>	186
5.5.1.3.	<i>Hủy Cookie</i>	187
5.5.2.	Tổng quan về session	187
5.5.2.1.	<i>Thiết lập session</i>	187
5.5.2.2.	<i>Sử dụng giá trị của session</i>	188
5.5.2.3.	<i>Hủy bỏ session</i>	188
CHƯƠNG 6		190
PHP & MYSQL		190
BÀI 1		190
THAO TÁC TRÊN CƠ SỞ DỮ LIỆU MYSQL		190
6.1.1.	Giới thiệu	190
6.1.2.	Đặc điểm	190
6.1.2.1.	<i>Tạo Cơ sở dữ liệu</i>	191
6.1.2.2.	<i>Xóa Cơ sở dữ liệu</i>	192
6.1.2.3.	<i>Thao tác với bảng trong MySQL</i>	192
6.1.2.4.	<i>Import và export dữ liệu</i>	197
BÀI 2		199
KẾT NỐI CƠ SỞ DỮ LIỆU MYSQL		199
6.2.1.	Các bước xây dựng chương trình có kết nối tới Cơ sở dữ liệu	199
6.2.2.	Thông báo lỗi	202
BÀI 3		203
THAO TÁC TRÊN CƠ SỞ DỮ LIỆU MYSQL VỚI PHP		203
6.3.1.	Đếm số lượng mẫu tin	203
6.3.2.	Hiển thị dữ liệu	203
6.3.3.	Lưu thông tin mới vào CSDL	210
6.3.4.	Cập nhật dữ liệu	210
6.3.5.	Xóa dữ liệu	210
CHƯƠNG 7		213
HOÀN THÀNH MỘT TRANG WEB		213
7.1.	Yêu cầu	213
7.2.	Phân tích	216
7.3.	Thiết kế	216
7.4.	Module dùng chung	218
7.5.	Các file dùng chung	218
7.6.	Các chức năng	219
PHỤ LỤC		232
CÔNG CỤ HỖ TRỢ THIẾT KẾ WEB		232
A.	Ý nghĩa và một số công cụ hỗ trợ thiết kế Web	232
B.	Một số chức năng hỗ trợ đáng chú ý	234
C.	Xuất bản kết quả	238
TÀI LIỆU THAM KHẢO		239

## DANH MỤC HÌNH

Hình 1-1-1: Mô hình hoạt động của dịch vụ Web .....	5
Hình 1-1-2: Cấu trúc trang HTML.....	7
Hình 1-1-3: Trang Vidu1-1-1.html trên trình duyệt Firefox .....	7
Hình 1-2-1: Ví dụ về canh lề đoạn văn bằng thẻ <div> .....	11
Hình 1-2-2: Trang Vidu1-2-2.html định dạng đoạn văn bằng thẻ <p> .....	12
Hình 1-2-3: Trang Vidu1-2-3.html định dạng tiêu đề bằng các thẻ Heading.....	13
Hình 1-2-4: Trang Vidu1-2-4.html – tổng hợp các thẻ định dạng văn bản.....	16
Hình 1-2-5: Trang Vidu1-2-5.html – Một số thử nghiệm với thẻ <marquee> .....	17
Hình 1-2-6: Trang Vidu1-2-6.html, một ví dụ về danh sách không thứ tự .....	19
Hình 1-2-7: Trang Vidu1-2-7.html, một ví dụ về danh sách có thứ tự .....	21
Hình 1-2-8: Trang Vidu1-2-8.html .....	23
Hình 1-2-9: Trang Vidu1-2-9.html .....	25
Hình 1-3-1: Liên kết trong.....	28
Hình 1-3-2: Liên kết ngoài .....	28
Hình 1-3-3: Liên kết có thể thực thi được .....	28
Hình 1-3-4: Trang Vidu1-3-1.html .....	29
Hình 1-3-5: Trang Vidu1-3-2.html .....	33
Hình 1-3-6: giaodienFrameset1.html .....	35
Hình 1-3-7: Trang frameset1_banner.html .....	36
Hình 1-3-9: Trang frameset1_nd1.html .....	36
Hình 1-3-10: Trang frameset1_nd2.html.....	37
Hình 1-3-11: Trang Vidu1-3-5.html .....	40
Hình 1-4-1: Trang Vidu1-4-1.html .....	50
Hình 1-4-2: Trang Vidu1-4-2.html .....	50
Hình 1-5-1: Trang Vidu1-5-1.html .....	67
Hình 2-1: Trang Vidu2-1.html.....	81
Hình 2-2: Trang Vidu2-2.html.....	83
Hình 2-3: Trang Vidu2-3.html.....	85
Hình 2-4: Trang Vidu2-4.html.....	94
Hình 2-5: Trang Vidu2-4.html khi đưa con trỏ qua liên kết CSS.....	95
Hình 3-1: Hộp thoại OK.....	107
Hình 3-2: Hộp thoại YES/NO.....	107
Hình 3-3: Hộp thoại lấy thông tin .....	107
Hình 4-2-1:Lựa chọn phiên bản WampServer để tải về.....	134
Hình 4-2-2: Bước 1 – quá trình cài đặt WampServer2 .....	135
Hình 4-2-3: Bước 2 - quá trình cài đặt WampServer2 .....	135
Hình 4-2-4: Bước 3 - quá trình cài đặt WampServer2 .....	136
Hình 4-2-5: Bước 4 - quá trình cài đặt WampServer2 .....	136
Hình 4-2-6: Bước 5 - quá trình cài đặt WampServer2 .....	137
Hình 4-2-7: Bước 6 - quá trình cài đặt WampServer2 .....	137
Hình 4-2-8: Bước 7 - quá trình cài đặt WampServer2 .....	138
Hình 4-2-9: Bước 8 - quá trình cài đặt WampServer2 .....	138
Hình 4-2-10: Bước 9 - quá trình cài đặt WampServer2 .....	139
Hình 4-2-11: Đổi cổng cho Apache .....	140
Hình 4-2-12: Khởi động lại các dịch vụ trong WampServer2.....	141
Hình 4-2-13: Giao diện quản trị WebServer bằng WampServer2 .....	141
Hình 4-2-14: Tạo một thư mục chứa Website với tên là giaotrinhtLTW .....	142

Hình 4-2-15: Duyệt thử trang web trên WampServer.....	143
Hình 5-3-1: Sử dụng lại mã PHP .....	169
Hình 5-3-2: Đọc file .....	173
Hình 5-3-3: Ghi dữ liệu xuống tập tin.....	174
Hình 5-4-1: Thể <form> .....	182
Hình 5-4-2: Ví dụ 5-4-1.....	182
Hình 6-1-1: Export dữ liệu .....	198
Hình 6-3-1: Hiển thị dữ liệu dạng bảng .....	206
Hình 6-3-2: Ví dụ 6-3-8 – Trang thông tin khách hàng .....	207
Hình 7-1: Trang Index.php .....	213
Hình 7-2: Trang Index.php khi đăng nhập quyền thành viên .....	214
Hình 7-3: Trang Index.php khi đăng nhập quyền quản trị .....	215
Hình 7-4: Trang đăng ký thành viên .....	220
Hình 7-5: Trang thông tin cá nhân .....	221
Hình 7-6: Trang tải sản phẩm .....	223
Hình 7-7: Trang quản trị người dùng .....	224
Hình 7-8: Trang sửa thông tin thành viên .....	225
Hình 7-9: Trang thêm sản phẩm .....	226
Hình 7-10: Trang quản trị nội dung sách .....	227
Hình 7-11: Trang sửa một đầu sách .....	228
Hình 7-12: Trang chọn bạn Chat .....	229
Hình 7-13: Trang Chat trực tuyến.....	230
Hình PL-1: Màn hình khởi động – chọn cửa sổ làm việc mà bạn mong muốn.....	233
Hình PL-2: Chọn kiểu file .....	233
Hình PL-3: Cửa sổ Code .....	234
Hình PL-4: Dreamweaver hỗ trợ code .....	234
Hình PL-5: Chế độ Split .....	235
Hình PL-6: Chế độ Design .....	235
Hình PL-7: Thanh Common .....	236
Hình PL-8: Thanh Layout .....	236
Hình PL-9: Thanh form.....	236
Hình PL-10: Thanh Text .....	237
Hình PL-11: Thanh Application .....	237
Hình PL-12: Hiệu chỉnh Encoding .....	237
Hình PL-13: Xuất bản kết quả .....	238

# CHƯƠNG 1

## NGÔN NGỮ HTML

### BÀI 1

#### **TỔNG QUAN VỀ INTERNET, WEB VÀ THIẾT KẾ WEB**

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Trình bày tổng quan về mạng Internet và Web
- Trình bày tổng quan về thiết kế Web

#### **1.1.1 Lịch sử phát triển của Internet và Web**

Tiền thân của Internet là ARPANET, mạng máy tính được xây dựng bởi Bộ Quốc phòng Mỹ (DOD) vào năm 1969 dùng để kiểm thử độ tin cậy của mạng và nhằm kết nối những cơ sở nghiên cứu với mục đích quân sự, bao gồm một số lượng lớn các trường đại học, viện nghiên cứu. ARPANET khởi đầu với quy mô nhỏ nhưng đã nhanh chóng bành trướng ra khắp nước Mỹ.

Thành công của ARPANET được nhân lên gấp bội, ngày càng nhiều trường đại học đăng ký tham gia. Tuy nhiên, quy mô lớn của mạng đã gây khó khăn trong vấn đề quản lý. Từ đó ARPANET được chia thành hai phần: MILNET là hệ thống mạng dành cho quân sự và ARPANET mới nhỏ hơn, không thuộc DOD. Tuy nhiên hai mạng vẫn liên kết với nhau nhờ giải pháp kỹ thuật gọi là IP (Internet Protocol – Giao thức Internet), cho phép thông tin truyền từ mạng này sang mạng khác khi cần thiết.

Vào đầu năm 1980, giao thức TCP/IP được phát triển và nhanh chóng trở thành giao thức chuẩn được dùng trên mạng ARPANET. Với công nghệ mới này, số lượng mạng máy tính đã tăng lên nhanh chóng. Mạng ARPANET ban đầu đã trở thành mạng đường trực (backbone) cho mạng máy tính chạy trên giao thức TCP/IP gồm hàng ngàn máy tính thuộc các mạng cục bộ khác nhau. Mạng máy tính này chính là mạng Internet.

Như vậy, Internet bao gồm nhiều hệ mạng dùng nhiều giao thức khác nhau. Các mạng máy tính này muốn liên lạc được với nhau thì cần có một giao thức hay một ngôn ngữ chung. Ngôn ngữ chung này chính là giao thức TCP/IP. Tất cả các mạng được nối vào Internet đều sử dụng IP.

Theo trang <http://info.cern.ch/>, năm 1989, Tim Berners Lee tại viện Kỹ thuật hạt nhân Châu Âu – CERN (Conseil Européen pour la Recherche Nucléaire) đã viết bản đề nghị đầu tiên để trình bày cơ hội nối siêu văn bản với Internet. Đến năm 1990, dựa trên những khái niệm siêu văn bản giống của hệ thống mẫu ENQUIRE mà ông đã xây dựng năm 1980, ông phát minh ra giao thức World Wide Web – viết tắt là WWW hay gọi ngắn gọn là Web, sau đó là ngôn ngữ HTML (HyperText Markup Language). Có thể nói đây là một cuộc cách mạng trên Internet vì người ta có thể truy cập và trao đổi thông tin một cách dễ dàng. Website đầu tiên được xây tại địa chỉ <http://info.cern.ch/> và được phát hành ngày 6 tháng 8 năm 1991. Nó giải thích World Wide Web là cái gì, cách tìm trình duyệt và cách thiết lập trình phục vụ Web.

Tuy nhiên Web lúc bấy giờ không có bất kỳ điểm nào giống như Web mà chúng ta biết ngày nay. Trình duyệt Web đầu tiên không thể điều khiển bằng chuột và không hỗ trợ đồ họa. Nhưng người dùng Internet lúc đó đã nhanh chóng cảm nhận được sức mạnh tiềm năng của Web.

Web trưởng thành rất nhanh. Trong vài năm ngắn ngủi đã có những cải tiến lớn. Ý nghĩa nhất là các trình duyệt Web dùng chuột và khả năng hỗ trợ đa phương tiện: ViolaWWW (1992), NCSA Mosaic 1.0 (1993), Netscape Navigator 1.0 (1994), Microsoft Internet Explorer 1.0 (1995), ...

### **1.1.2 Các khái niệm cơ bản trên Internet và Web**

Giao thức TCP/IP (Transmission Control Protocol / Internet Protocol): còn gọi là cơ chế truyền thông tin trên Internet. Đây là bộ giao thức được sử dụng nhiều nhất gồm hơn 100 giao thức cho phép truyền dữ liệu giữa các máy tính trên Internet đồng thời có nhiệm vụ định địa chỉ (address) và vạch đường (routing).

Giao thức WWW (Word Wide Web): hay gọi ngắn gọn là Web. Ngày nay, WWW được xem là một dịch vụ quan trọng của Internet cho phép tra cứu tài nguyên thông tin, trao đổi thông tin, thực hiện giao dịch kinh doanh,... trên Internet.

Siêu văn bản (HyperText): được hiểu là loại văn bản mà khi muốn đọc thêm thì sẽ có cơ chế “nở ra” vì có những tham chiếu tới những khái niệm hay văn bản khác, do nhà tin học Ted Nelson đưa ra từ năm 1965 và được tin học hóa bởi kỹ sư trẻ người Anh – Tim Berners Lee. Nó trở thành hệ thống liên kết các phần tử thông tin nhè vào các liên kết có thể kích hoạt hay còn gọi là siêu liên kết (Hyperlink). Các phần tử thông tin trong siêu văn bản gồm:

Dữ liệu dạng văn bản (Text).

Dữ liệu dạng đa phương tiện (Mutimedia) như âm thanh, hình ảnh.

Siêu văn bản khác: bên trong tài liệu siêu văn bản có thể chứa các liên kết đến các phần khác bên trong tài liệu đó hoặc đến các tài liệu bên ngoài khác.

Những chương trình thực thi được: được viết bằng các ngôn ngữ lập trình như Java, JavaScript, PHP Script, ASP,...

Ngôn ngữ HTML (HyperText Markup Language): ngôn ngữ đánh dấu siêu văn bản. Đây là ngôn ngữ dùng để tạo ra các trang siêu văn bản theo mô tả của Tim Berners Lee.

Siêu liên kết (Hyperlink): là mối nối kết giữa hai phần tử thông tin trong một siêu văn bản.

Trình phục vụ Web (Web Server): là chương trình được cài đặt tại máy chủ (Server), cung cấp dịch vụ Web, lắng nghe, phân tích các yêu cầu về Web, thực hiện tìm kiếm thông tin và trả kết quả về cho người dùng Web.

Trình duyệt Web (Web Browser): là chương trình được cài đặt trên máy người dùng (Client), dùng để gửi yêu cầu tra cứu thông tin đến trình phục vụ Web, nhận và hiển thị kết quả do trình phục vụ Web gửi về máy người dùng.

Giao thức HTTP (HyperText Transfer Protocol): là giao thức ở tầng ứng dụng (Application Layer), là giao thức dùng để giao tiếp giữa Web Browser và Web Server. HTTP sử dụng giao thức vận chuyển TCP.

Dịch vụ FTP (File Transfer Protocol): là dịch vụ cho phép truyền và nhận file giữa Client và Server.

URL (Uniform Resource Locator): là địa chỉ dùng để định vị nguồn tài nguyên trong dịch vụ Web. Mỗi tài liệu và tập tin trên Internet đều có một URL riêng. Cú pháp tổng quát của một URL có dạng:

<Giao thức>://<Tên miền>[:<cổng>]/<Đường dẫn>/<Tên file>

Ví dụ về URL:

<http://joomlaviet.org/forum/index.php>

<http://tvu.edu.vn>

[http://ktcn.tvu.edu.vn/index.php?option=com\\_content&view=frontpage](http://ktcn.tvu.edu.vn/index.php?option=com_content&view=frontpage)

Địa chỉ định vị nguồn tài nguyên phân ra là hai loại: địa chỉ tuyệt đối và địa chỉ tương đối. Địa chỉ tuyệt đối là một địa chỉ có cú pháp đầy đủ gồm giao thức, tên miền, tên đường dẫn (nếu có) và tên tập tin Web đang hiển thị. Địa chỉ tương đối chỉ gồm đường dẫn (nếu có) và tên tập tin. Trình duyệt sẽ tự động xác định địa chỉ tuyệt đối bằng cách lấy phần thông tin bị khuyết từ URL của trang Web hiện hành kết hợp với URL tương đối.

Trong thực tế, ứng dụng Web luôn tồn tại hai loại là trang Web tĩnh và trang Web động. Trang Web tĩnh là trang HTML không kết nối cơ sở dữ liệu. Ngược lại, trang Web động là trang Web có kết nối cơ sở dữ liệu. Điều này có nghĩa là mỗi khi trang Web động được “làm tươi”, dữ liệu trình bày trên trang Web đó được đọc từ cơ sở dữ liệu.

Nói cách khác, cho dù đó là trang Web tĩnh hay động, nếu bạn muốn trình bày dữ liệu trên trình duyệt Web, bạn cần phải khai báo các thẻ HTML bên trong theo một quy luật nhất định.

Mỗi trang Web được gọi là Webpage. Tập hợp nhiều trang Web được liên kết với nhau theo một cấu trúc nào đó do tổ chức hoặc cá nhân xây dựng nên được gọi là một Website. Mỗi Website sẽ có một trang Web được hiển thị đầu tiên khi người dùng truy cập vào Website đó, trang Web đầu tiên này được gọi là Homepage. Homepage cung cấp cái nhìn tổng quan về Website.

### 1.1.3 Các dịch vụ cơ bản trên Internet

a. Dịch vụ thư điện tử (Mail Service): Bạn có thể gửi hoặc nhận các thư điện tử (Electronic Mail- Email) từ bất cứ một nơi nào với điều kiện là người nhận và người gửi phải có một địa chỉ Email và máy tính sử dụng có nối mạng Internet. Việc sử dụng hộp thư điện tử sẽ giúp bạn rút ngắn thời gian cũng như quy trình gửi và nhận thư theo cách truyền thống trước đây rất nhiều.

b. Dịch vụ tin điện tử (News): Cho phép trao đổi các bài báo và bản tin điện tử (Bulletin) xuyên suốt Internet. Các bài báo được đặt tại trung tâm cơ sở dữ liệu trên suốt không gian Internet. Người dùng truy cập vào cơ sở dữ liệu đó để lấy về các bài báo họ cần. Điều này làm giảm tắc nghẽn trên mạng và loại trừ việc lưu trữ nhiều bản sao không cần thiết của mỗi bài báo trên từng hệ thống của người dùng.

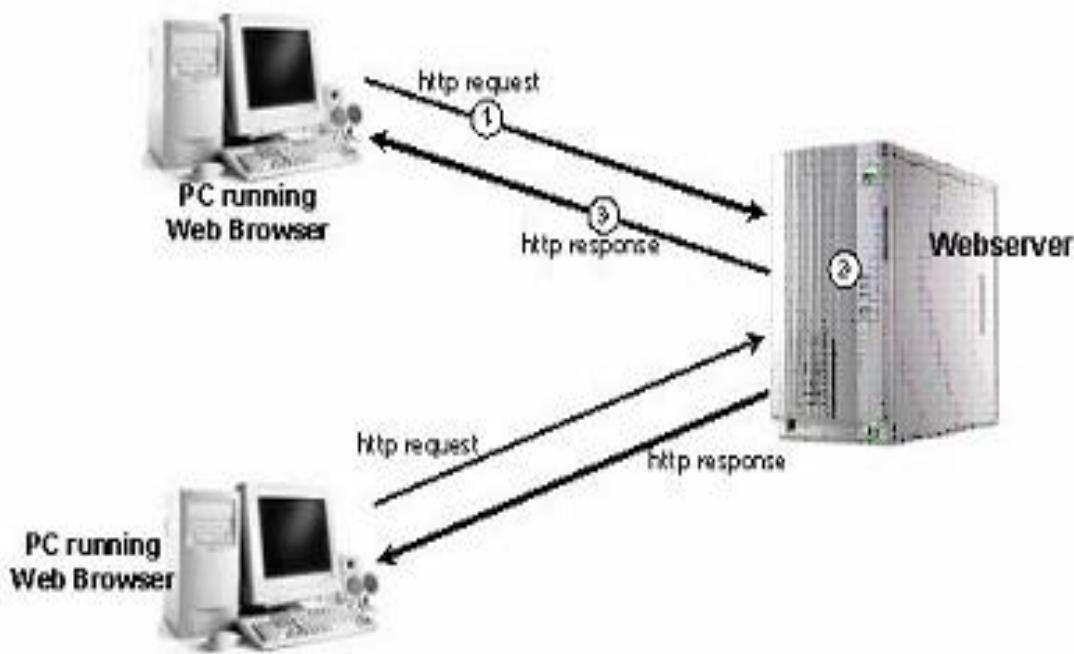
c. Dịch vụ truyền tập tin (FTP: File Transfer Protocol): FTP là một hệ thống chính

yêu để chuyển tải File giữa các máy vi tính vào Internet. File được chuyển tải có dung lượng rất lớn. FTP hầu hết được sử dụng cho việc chuyển tải những dữ liệu mang tính cá nhân. Thông thường, bạn sẽ dùng FPT để chép File trên một máy chủ từ xa vào máy của bạn, việc xử lý này gọi là nạp xuống (downloading). Tuy nhiên, bạn cũng có thể chuyển File từ máy của bạn vào máy chủ từ xa, điều này gọi là nạp lên (uploading). Thêm vào đó, nếu cần thiết, FTP sẽ cho phép bạn chép File từ một máy chủ từ xa tới một máy khác.

d. Dịch vụ Web (World Wide Web – WWW): Đây là khái niệm mà người dùng Internet quan tâm nhiều nhất hiện nay. Người ta viết tắt là WWW hay gọi ngắn gọn là Web. Khác với dịch vụ Web trước đây, Web ngày nay chứa thông tin bao gồm văn bản, hình ảnh, âm thanh và thậm chí cả video được kết hợp với nhau. Người dùng cần sử dụng một trình duyệt Web để truy cập vào các Website trên Internet. Hiện nay, có rất nhiều trình duyệt thông dụng nhất là Firefox - miễn phí, là sản phẩm của Mozilla, Chrome được phát triển miễn phí bởi Google, Internet Explorer của hãng Microsoft, Netscape của hãng Sun, Opera của Opera Software, Safari của Apple,...

#### 1.1.4 Cách hoạt động của một trang Web

Cách hoạt động của một trang Web thông qua mô hình sau:



**Hình 1-1-1: Mô hình hoạt động của dịch vụ Web**

Giải thích mô hình: Khi người sử dụng Internet mở một trình duyệt (Web Browser) trên máy tính, gõ vào thanh địa chỉ của trình duyệt một địa chỉ của một trang Web và nhấn

phím enter để xác nhận lệnh. Điều đó có nghĩa là một yêu cầu về dịch vụ HTTP đã phát sinh và được gửi đến trình phục vụ Web ①. Trình phục vụ Web khi nhận được yêu cầu sẽ tiến hành tìm kiếm trên hệ thống mà Web Server đang lưu trữ địa chỉ của trang đang được yêu cầu ②. Kết quả tìm kiếm sẽ được gửi về cho máy tính mà người sử dụng đã gửi yêu cầu ③. Nếu tìm thấy, trình duyệt Web trên máy tính của người sử dụng sẽ có nhiệm vụ hiển thị nội dung trang Web theo ý định của người thiết kế trang Web đó.

### 1.1.5 Cấu trúc trang HTML

Như đã trình bày ở trên, muốn trình bày và định dạng dữ liệu trên trình duyệt phải theo cấu trúc HTML. Đây là một ngôn ngữ thông dụng dùng để thiết kế Web tĩnh. Một trang HTML sẽ được đánh dấu bắt đầu bằng thẻ `<HTML>` và kết thúc bởi thẻ `</HTML>`. Như vậy, các nội dung đặt ngoài cặp thẻ này đều không hợp lệ. Thẻ `<HTML>` được gọi là thẻ mở, thẻ `</HTML>` được gọi là thẻ đóng. Bên trong hai thẻ này, bạn có thể khai báo và sử dụng hầu hết các thẻ HTML. Tuy nhiên có một vài thẻ HTML không được hỗ trợ bởi trình duyệt này nhưng lại được hỗ trợ bởi trình duyệt khác. Những trường hợp đặc biệt đó, nhóm biên soạn tài liệu chúng tôi sẽ giới thiệu đến bạn khi chúng ta tìm hiểu từng thẻ HTML cụ thể.

Trang Web tĩnh sẽ có tên mở rộng là `.html` hoặc `.htm`. Có thể soạn thảo mã HTML bằng bất kỳ chương trình soạn thảo văn bản nào. Ví dụ, bạn có thể soạn bằng Microsoft Office Word, Notepad, WordPad, FrontPage, Dreamweaver, ...

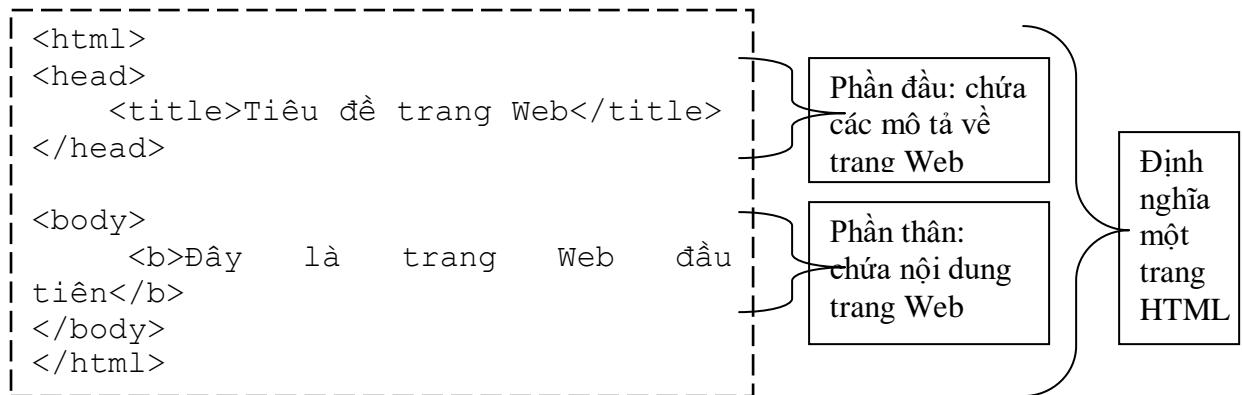
Ngôn ngữ HTML không phân biệt ký tự HOA / thường khi khai báo tên thẻ. Ví dụ: khi bạn gõ `<html>` hay `<Html>` hay `<HTML>` đều mang ý nghĩa như nhau là khai báo bắt đầu trang HTML. Và khi bạn khai báo một thẻ đánh dấu phần đầu trang HTML là `<head>` thì các thẻ `</head>` hay `</Head>` hay `</HEAD>` đều là thẻ đóng của nó.

Các thẻ HTML đã được định nghĩa sẵn, bạn chỉ cần nhớ tên thẻ và ý nghĩa của nó để sử dụng mà không cần tự định nghĩa cho bất kỳ thẻ nào cả. Điều quan trọng là bạn cần nhớ các thuộc tính đi kèm bên trong thẻ để tùy chỉnh cho phần nội dung mà bạn muốn hiển thị lên trình duyệt sao cho linh hoạt và đúng ý định hiển thị của bạn. Một điều lưu ý nữa là bạn cần gõ chính xác tên thẻ và không có bất kỳ dấu cách nào trong tên các thẻ HTML đã được định nghĩa sẵn. Ví dụ: các cách viết tên thẻ sau đây đều sai: `< head>`, `<h ead>`, `<he ad>`, `<hea d>`. Nếu bạn muốn thêm thuộc tính cho thẻ thì sau tên thẻ, bạn nhấn phím khoảng cách sau đó xác định thuộc tính cho thẻ.

Thẻ HTML có hai loại: thẻ kép và thẻ đơn. Thẻ kép là loại thẻ có thẻ mở và thẻ đóng.

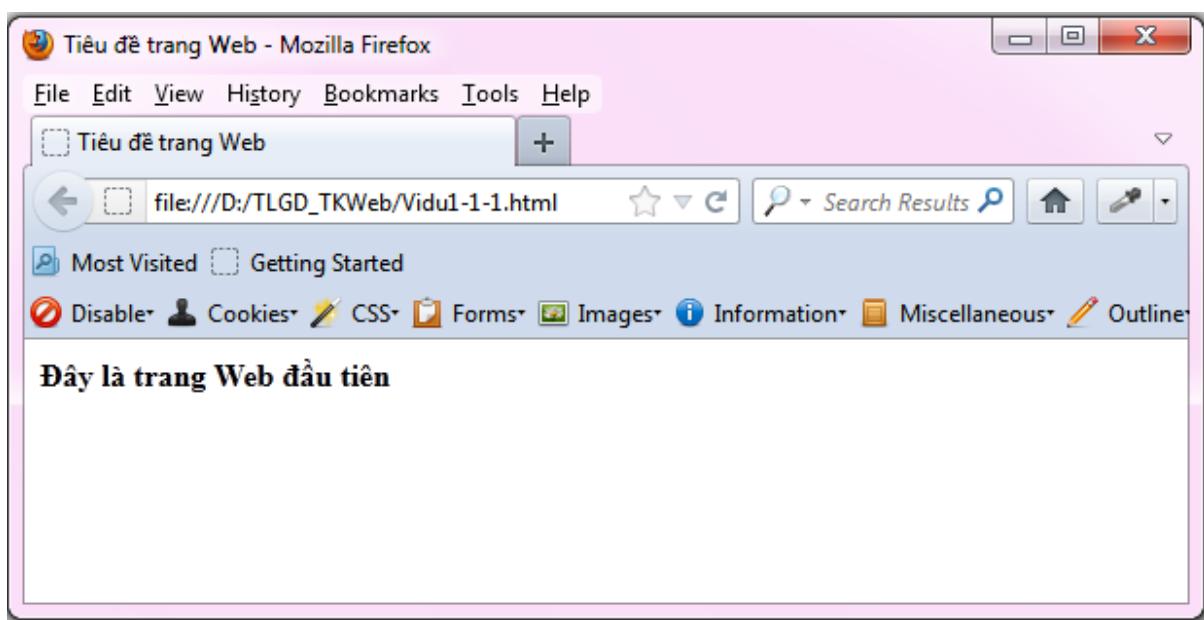
Thẻ mở được viết như sau: <tênthẻmở>. Thẻ đóng phải được khai báo khi đã có thẻ mở. Cách viết thẻ đóng như sau: </tênthẻmở>. Nội dung cần hiển thị lên Web, bạn đặt giữa cặp thẻ mở và thẻ đóng để có được định dạng của thẻ đã khai báo. Thẻ đơn là loại thẻ chỉ có thẻ mở mà không có thẻ đóng.

Nếu sử dụng nhiều thẻ HTML để định dạng cho cùng một nội dung thì các thẻ đó phải lồng nhau. Tức là thẻ HTML tuân theo quy luật mở trước thì đóng sau.



Hình 1-1-2: Cấu trúc trang HTML

Để soạn thảo mã HTML, ta Notepad gõ và lưu nội dung trang HTML trong **Hình 1-1-2** với tên **Vidu1-1-1.html**. Lưu ý: để hiển thị được tiếng Việt lên trình duyệt, trong quá trình lưu tập tin, bạn cần chọn **Encoding: Unicode** hoặc **Encoding: UTF-8**. Sau đó mở tập tin này bằng trình duyệt Web, ta sẽ thấy giao diện của trang như sau:



Hình 1-1-3: Trang Vidu1-1-1.html trên trình duyệt Firefox

Bạn có thể thêm thuộc tính định dạng cho thẻ HTML với cấu trúc:

```
<tênthẻ thuộctính_1="giátrị_1" thuộctính_2="giátrị_2" ... thuộctính_n="giátrị_n">
```

❖ Câu hỏi (bài tập) củng cố:

1. Khi một trang Web có sử dụng rất nhiều hình ảnh động trên giao diện. Vậy có thể khẳng định đó là một trang Web động không? Tại sao?
2. Trình bày sơ lược lịch sử phát triển của www.
3. Trình bày chức năng của Web Browser và Web Server.

## BÀI 2

### **CÁC THẺ HTML CƠ BẢN**

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Vận dụng các thẻ HTML cơ bản để thiết kế trang Web

#### **1.2.1 Thẻ <body>, chú thích và các ký hiệu đặc biệt**

Trong cặp thẻ `<body> ... </body>` chúng ta có thể dùng các thuộc tính định dạng cho toàn nội dung của trang như:

*Background=“URL”* với URL là đường dẫn đến tập tin hình ảnh dùng làm ảnh nền cho phần nội dung của trang.

*Bgcolor= “color”* với *color* là mã màu ở hệ thập lục phân hoặc các từ khóa tên màu bằng tiếng Anh như: red (màu đỏ), blue (màu xanh da trời), green (màu xanh lá), white (màu trắng), .... để định dạng màu nền trang.

*Text=“color”* với *color* là mã màu ở hệ thập lục phân hoặc các từ khóa tên màu bằng tiếng Anh như: red (màu đỏ), blue (màu xanh da trời), green (màu xanh lá), white (màu trắng), .... để định dạng màu chữ cho toàn trang.

Để ghi chú một đoạn mã HTML, ta đặt nội dung chú thích trong cặp thẻ `<!-- và -->`. Khi dịch trang Web, trình duyệt sẽ bỏ qua phần nội dung nằm trong cặp thẻ này.

Trong ngôn ngữ HTML, một số ký tự đặc biệt bạn không thể gõ từ bàn phím. Ví dụ nếu chỉ gõ khoảng cách giữa các từ là một ký tự khoảng trắng hay nhiều ký tự khoảng trắng thì kết quả hiển thị lên trình duyệt là như nhau. Vì vậy, nếu bạn muốn có nhiều khoảng trắng liên tục thì phải thực hiện gọi mã HTML cho ký hiệu đặc biệt. Sau đây là bảng liệt kê một số ký hiệu đặc biệt thường dùng:

MÃ HTML	Ý NGHĨA	MÃ HTML	Ý NGHĨA
&ampnbsp	Khoảng trắng	&trade;	TM
&lt;	<	&copy;	©

MÃ HTML	Ý NGHĨA	MÃ HTML	Ý NGHĨA
&gt;	>	&reg;	®
&	&	&yen;	¥
&quot;	“	&euro;	€
&sum;	Σ	&frac12;	½

## 1.2.2 Thẻ định dạng văn bản

### 1.2.2.1 Thẻ *<div> ... </div>*

Cặp thẻ này cho phép bạn định dạng cho đoạn văn bằng các thuộc tính bên trong thẻ. Tuy nhiên, cặp thẻ này không kèm theo ký tự xuống dòng cho đoạn văn. Ví dụ, bạn muốn canh lề cho đoạn văn, bạn có thể sử dụng thuộc tính *align* của thẻ *<div>*. Giá trị của thuộc tính *align* có bốn giá trị: *center / justify / left / right*.

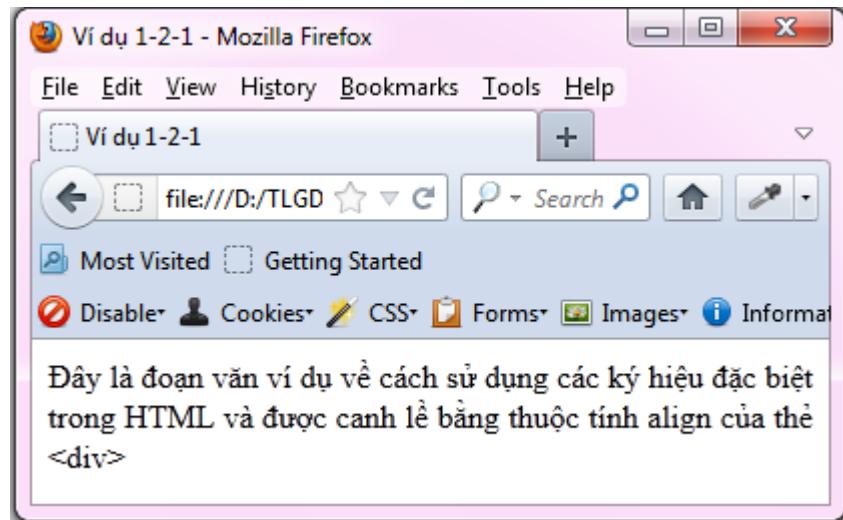
Ví dụ 1-2-1: Canh lề cho đoạn văn bằng thẻ *<div>*

```

|<html>
|<head>
|<meta http-equiv="Content-Type" content="text/html;
| charset=utf-8" />
|<title>Ví dụ 1-2-1</title>
|</head>
|
|<body>
|    <div align="justify">Đây là đoạn văn ví dụ về cách sử
|    dụng các ký hiệu đặc biệt trong HTML và được canh lề bằng
|    thuộc tính align của thẻ &lt;div&gt;</div>
|</body>
|</html>

```

Giải thích: Một cách khác để hiển thị được tiếng Việt lên trình duyệt, ta dùng thẻ *<meta>* với các thuộc tính như trong ví dụ 1-2-1. Lưu ví dụ trên với tên **Vidu1-2-1.html** và mở tập tin bằng trình duyệt, ta được:



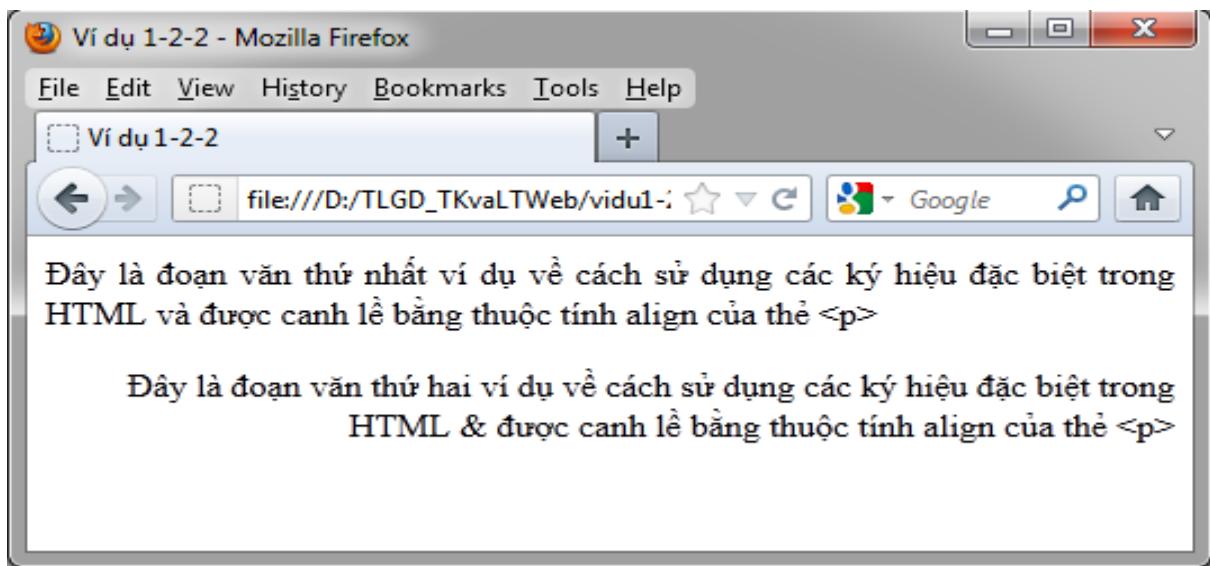
**Hình 1-2-1: Ví dụ về canh lề đoạn văn bằng thẻ <div>**

#### 1.2.2.2 Thẻ <p> ... </p>

Tương tự thẻ <div>, thẻ <p> cũng được dùng để định dạng đoạn văn. Nhưng khác với thẻ <div>, thẻ <p> ngoài cho phép canh chỉnh lề cho đoạn văn, khi kết thúc đoạn được đánh dấu bằng thẻ </p> dữ liệu trình bày sẽ tự động xuống dòng. Thẻ đoạn <p> cũng có thuộc tính align với bốn giá trị center / justify / left / right.

Ví dụ 1-2-2: Thẻ định dạng đoạn <p>

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Ví dụ 1-2-2</title>
</head>
<body>
    <p align="justify">Đây là đoạn văn thứ nhất ví dụ về
    cách sử dụng các ký hiệu đặc biệt trong HTML và được canh lề
    bằng thuộc tính align của thẻ &lt;p&gt;</p>
    <p align="right">Đây là đoạn văn thứ hai ví dụ về cách sử
    dụng các ký hiệu đặc biệt trong HTML &copy; và được canh lề
    bằng thuộc tính align của thẻ &lt;p&gt;</p>
</body>
</html>
```



**Hình 1-2-2: Trang Vidu1-2-2.html định dạng đoạn bằng thẻ <p>**

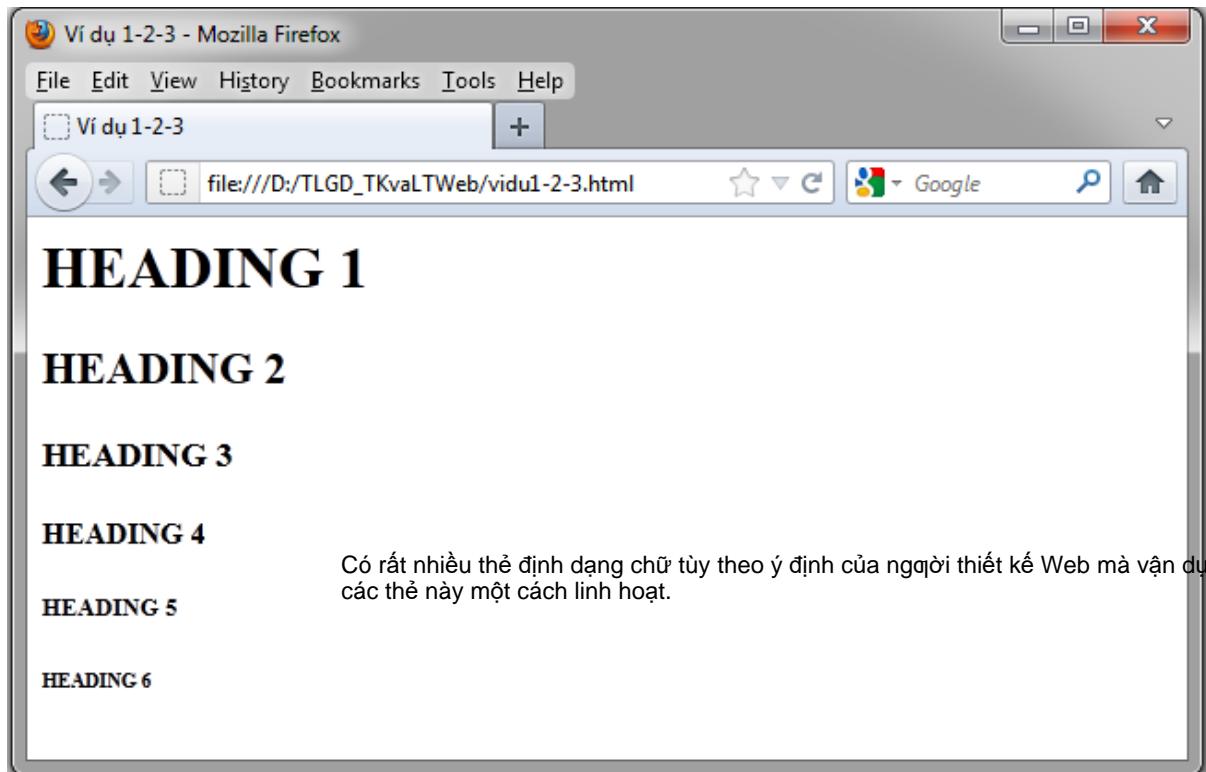
### **1.2.2.3 Các thẻ định dạng tiêu đề**

HTML định nghĩa sáu cấp độ của tiêu đề: `<h1> ... </h1>`, `<h2> ... </h2>`, ..., `<h6> ... </h6>` với độ lớn giảm dần từ h1 đến h6. Kết thúc thẻ tiêu đề, nội dung trình bày cũng tự động xuống dòng. Chúng ta có thể tham khảo ví dụ 1-2-3 bên dưới để hiểu rõ hơn về các thẻ tiêu đề.

Ví dụ 1-2-3: Ví dụ về các thẻ Heading

```
-----  
|<html>  
|<head>  
|<meta http-equiv="Content-Type" content="text/html;  
| charset=utf-8" />  
|<title>Ví dụ 1-2-3</title>  
|</head>  
|<body>  
|  <H1> HEADING 1 </H1>  
|  <H2> HEADING 2 </H2>  
|  <H3> HEADING 3 </H3>  
|  <H4> HEADING 4 </H4>  
|  <H5> HEADING 5 </H5>  
|  <H6> HEADING 6 </H6>  
|</body>  
</html>  
-----
```

Dùng Notepad soạn thảo ví dụ trên và mở tập tin bằng trình duyệt, ta sẽ được giao diện như sau:



**Hình 1-2-3: Trang Vidu1-2-3.html định dạng tiêu đề bằng các thẻ Heading**

#### **1.2.2.4 Thẻ định dạng chữ**

Có rất nhiều thẻ định dạng chữ tùy theo ý định của người thiết kế Web mà vận dụng các thẻ này một cách linh hoạt.

Khi hiển thị nội dung lên Web, phím enter sẽ không có tác dụng. Vì vậy bạn có thể dùng thẻ *<br>* để xuống hàng. Đây là một thẻ đơn và không có thuộc tính bên trong.

Nếu bạn muốn kẽ một đường ngang trên giao diện Web, HTML hỗ trợ cho bạn thẻ *<hr>*. Đây cũng là một thẻ đơn. Bạn có thể trang trí cho đường kẽ này một số thuộc tính như: màu sắc (dùng thuộc tính *color*), độ rộng cho đường kẽ nếu trường hợp bạn không muốn kẽ đường ngang hết trang hãy sử dụng thuộc tính *width*, độ dày của đường kẽ được chỉ định trong thuộc tính *size* và cuối cùng là canh lề cho đường ngang với thuộc tính *align*.

Khi muốn hiển thị lên Web phần văn bản in đậm, bạn dùng cặp thẻ *<b> ... </b>*.

Khi muốn hiển thị lên Web phần văn bản in nghiêng, bạn dùng cặp thẻ *<i> ... </i>*.

Khi muốn hiển thị lên Web phần văn bản có gạch chân, bạn dùng cặp thẻ *<u> ... </u>*.

Khi muốn nhấn mạnh một nội dung nào đó trên Web, bạn có thể dùng cặp thẻ

`<strong> ... </strong>`, nội dung được nhấn mạnh sẽ được in đậm như khi dùng thẻ `<b>`.

Khi muốn làm nổi bật một nội dung nào đó trên Web, bạn có thể dùng cặp thẻ `<em> ... </em>`, nội dung được làm nổi bật sẽ được in nghiêng như khi dùng thẻ `<i>`.

Khi muốn hiển thị chữ lớn mà không quan tâm đến kích cỡ chữ, bạn có thể dùng cặp thẻ `<big> ... </big>`.

Tương tự, nếu muốn hiển thị chữ nhỏ mà không quan tâm đến kích cỡ của chữ, bạn có thể dùng cặp thẻ `<small> ... </small>`.

Cặp thẻ `<font> ... </font>` sẽ cho phép bạn định dạng phông chữ với thuộc tính `face`, màu chữ với thuộc tính `color`, cỡ chữ với thuộc tính `size`. Trong đó, giá trị chỉ định cho thuộc tính `face` là các tên phông. Giá trị của thuộc tính `color` là mã màu ở hệ thập lục phân hoặc các từ khóa tên màu bằng tiếng Anh như: red (màu đỏ), blue (màu xanh da trời), green (màu xanh lá), white (màu trắng), .... Giá trị của thuộc tính `size` là tăng dần từ 1 đến 7.

Nếu nội dung được đặt trong cặp thẻ `<pre> ... </pre>` thì sẽ được giữ nguyên định dạng như lúc soạn thảo. Điều này có nghĩa các phím enter hay khoảng trắng sẽ được hiểu như đúng ý nghĩa của nó khi dùng trong thẻ `<pre>`.

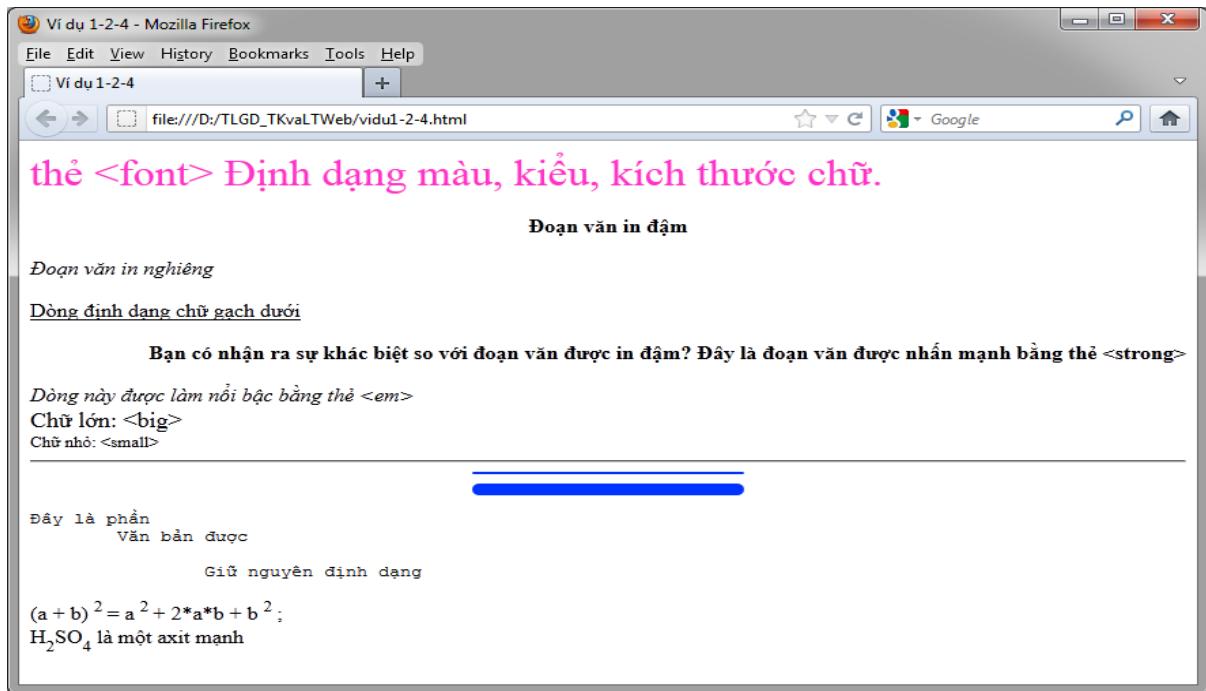
Trường hợp bạn muốn viết chỉ số trên, ví dụ như  $x^2$ ; hoặc chỉ số dưới, ví dụ như  $H_2$ . Trong trường hợp này bạn dùng cặp thẻ `<sup> ... </sup>` đối với chỉ số trên và cặp thẻ `<sub> ... </sub>` cho chỉ số dưới.

Sau đây là một ví dụ tổng hợp kiến thức về các thẻ định dạng chữ.

#### Ví dụ 1-2-4: Các thẻ định dạng chữ

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Ví dụ 1-2-4</title>
</head>
<body>
    <font face="Time New Roman" size="6" color="#FF33CC">
thẻ &lt;font&gt; Định dạng màu, kiểu, kích thước chữ.</font>
    <p align="center"><b>Đoạn văn in đậm</b></p>
    <p align="left"><i> Đoạn văn in nghiêng</i></p>
    <u> Dòng định dạng chữ gạch dưới</u><br />
    <p align="right"><strong>Bạn có nhận ra sự khác biệt so
với đoạn văn được in đậm? Đây là đoạn văn được nhấn mạnh bằng
thẻ &lt;strong&gt;</strong></p>
    <em> Dòng này được làm nổi bật bằng thẻ &lt;em&gt;;
</em><br />
    <big> Chữ lớn: &lt;big&gt; </big><br />
    <small> Chữ nhỏ: &lt;small&gt; </small>
    <hr>
    <hr color="#0033FF" width="200">
    <hr color="#0033FF" width="200" size="10">
    <pre>
Đây là phần
    Văn bản được

        Giữ nguyên định dạng
    </pre>
    (a + b) 2 = a 2 + 2*a*b + b 2
    ;
    <br />
    H2SO4 là một axit mạnh
    </body>
</html>
```



**Hình 1-2-4: Trang Vidu1-2-4.html – tổng hợp các thẻ định dạng văn bản**

### 1.2.2.5 Thẻ <marquee>

Nội dung đặt giữa cặp thẻ `<marquee>` và `</marquee>` sẽ chuyển động như kiểu quảng cáo trên các bảng điện tử. Các thuộc tính trong thẻ `<marquee>` gồm:

`Bgcolor="color"`: tô màu nền cho vùng hiển thị lệnh marquee.

`Direction="left / right / up / down"`: chỉ định hướng chuyển động cho chữ. Mặc định chữ sẽ chạy sang trái (`direction="left"`).

`Behavior="scroll / slide / alternate"`: thiết lập kiểu chạy chữ. *Scroll*: di chuyển tuần tự, đây là dạng mặc định. *Slide*: di chuyển cho đến khi gặp biên thì dừng lại. *Alternate*: di chuyển gặp biên thì đổi hướng.

`Title="nội dung"`: một đoạn nội dung sẽ được hiển thị khi người dùng di chuyển chuột vào vùng được định dạng marquee.

`Scrollmount="số"`: điều chỉnh tốc độ chạy, số càng lớn tốc độ chạy càng nhanh.

`Scrolldelay="số"`: chỉ định thời gian chờ sau mỗi lần cuộn, số càng lớn tốc độ càng chậm. Tính bằng đơn vị là giây.

`Loop="số"`: chỉ định số lần lặp lại của dòng marquee. Mặc định `Loop="-1"`, tức là lặp đến khi nào bạn tắt trang Web đó.

*Width=* “số”: chỉ định độ rộng của vùng chữ chạy. Có thể dùng đơn vị pixel hoặc phần trăm.

*Height=* “số”: chỉ định chiều cao của vùng chữ chạy. Có thể dùng đơn vị pixel hoặc phần trăm.

Ví dụ 1-2-5: định dạng chữ chạy

```
-->-----<-----  
<html>  
<head>  
<meta http-equiv="Content-Type" content="text/html;  
charset=utf-8" />  
<title>Ví dụ 1-2-5</title>  
</head>  
<body>  
    <marquee bgcolor="#999999" scrollamount="5"  
scrolldelay="10" direction="up" behavior="scroll" width="80%"  
height="100">Chào mừng các bạn đến với môn học Thiết kế  
Web</marquee>  
</body>  
</html>
```



**Hình 1-2-5: Trang Vidu1-2-5.html – Một số thử nghiệm với thẻ <marquee>**

Trong Ví dụ 1-2-5, chúng tôi tùy chỉnh cho vùng chữ chạy có màu nền là màu #999999, tốc độ chạy là 5, thời gian chờ là 10, hướng chữ chạy từ dưới lên, chạy theo dạng cuộn, chiều rộng chiếm 80% giao diện trang và chiều cao là 100px.

### 1.2.3 Thẻ định dạng danh sách

Cũng như một chương trình soạn thảo văn bản bình thường, Web cũng có khả năng hiển thị văn bản dạng danh sách. Ngôn ngữ HTML hỗ trợ cả hai loại danh sách không thứ tự

và có thứ tự.

#### 1.2.3.1 Danh sách không thứ tự `<ul>`

Đối với dạng danh sách không thứ tự, HTML hỗ trợ các dạng như *circle* (○), *disc* (●), *square* (■)

Cú pháp:

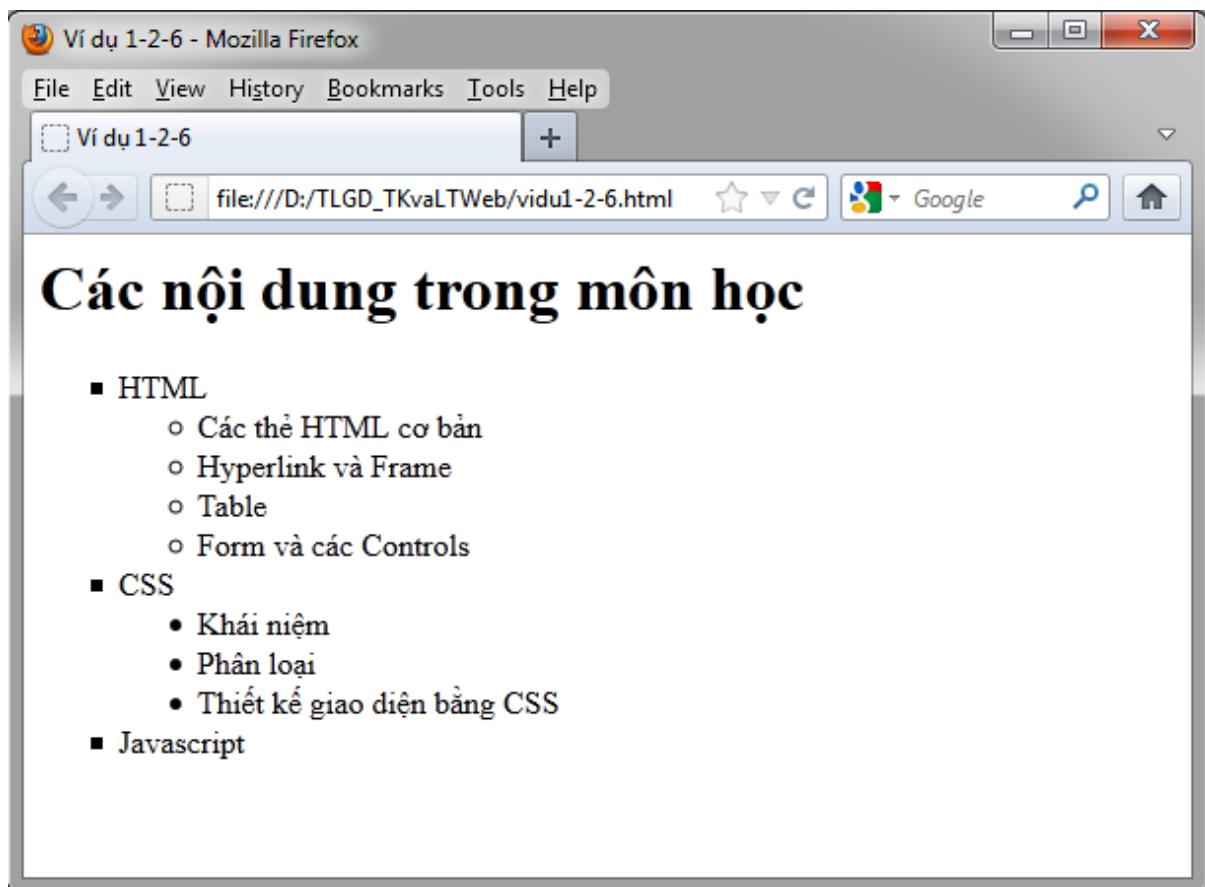
```
<ul type="circle/disc/square">
    <li> Mục thứ 1 </li>
    <li> Mục thứ 2 </li>
    ...
    <li> Mục thứ n </li>
</ul>
```

Ví dụ 1-2-6: Danh sách không thứ tự

```
-----<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Ví dụ 1-2-6</title>
</head>
<body>
    <h1>Các nội dung trong môn học</h1>
    <ul type="square">
        <li>HTML
            <ul type="circle">
                <li>Các thẻ HTML cơ bản</li>
                <li>Hyperlink và Frame</li>
                <li>Table</li>
                <li>Form và các Controls </li>
            </ul>
        </li>
        <li>CSS
            <ul type="disc">
                <li>Khái niệm</li>
                <li>Phân loại</li>
                <li>Thiết kế giao diện bằng CSS</li>
            </ul>
        </li>
        <li>Javascript</li>
    </ul>
</body>
</html>
```

Thông thường, người ta sẽ kết hợp các danh sách lại với nhau để được một menu phân cấp như trong Ví dụ 1-2-6. Cấu trúc menu phân cấp cần xác định mở và đóng đúng thẻ, đúng vị trí. Trong chương sau, chúng ta sẽ tiếp tục dùng CSS để hỗ trợ tạo menu động dạng DropDownList và FlyOut dựa trên thẻ danh sách không thứ tự này.

Một danh sách được xác định là con của một phần tử trong danh sách khác khi và chỉ khi cặp thẻ khai báo `<ul> ... </ul>` của danh sách con phải được nằm trọn trong cặp thẻ `<li> ... </li>` của phần tử trong danh sách khác đóng vai trò là cha.



Hình 1-2-6: Trang Vidu1-2-6.html, một ví dụ về danh sách không thứ tự

#### 1.2.3.2 Danh sách có thứ tự

Đối với dạng danh sách có thứ tự, HTML hỗ trợ các dạng như số (1, 2, 3, ...), ký tự chữ thường (a, b, c, ...), ký tự chữ hoa (A, B, C, ...), số La Mã thường (i, ii, iii, ...), số La Mã hoa (I, II, III, ...).

Cú pháp:

```

<ol type="1/a/A/i/I" start="n">
    <li> Mục thứ 1 </li>
    <li> Mục thứ 2 </li>
    ...
    <li> Mục thứ n </li>
</ol>

```

Khi muốn danh sách bắt đầu từ thứ tự lớn hơn thứ tự đầu tiên thì bạn nhập vào một số tự nhiên thể hiện thứ tự bắt đầu xuất hiện. Trường hợp không gọi thuộc tính *start* thì mặc định *start=“1”*.

#### Ví dụ 1-2-7: Danh sách có thứ tự

```

-----+
| <html>
| <head>
| <meta http-equiv="Content-Type" content="text/html;
| charset=utf-8" />
| <title>Ví dụ 1-2-7</title>
| </head>
| <body>
|     <h1>Các nội dung trong môn học</h1>
|     <ol type="I" start="3">
|         <li>HTML
|             <ol type="a" start="2">
|                 <li>Các thẻ HTML cơ bản</li>
|                 <li>Hyperlink và Frame</li>
|                 <li>Table</li>
|                 <li>Form và các Controls </li>
|             </ol>
|         </li>
|         <li>CSS
|             <ol type="1">
|                 <li>Khái niệm</li>
|                 <li>Phân loại</li>
|                 <li>Thiết kế giao diện bằng CSS</li>
|             </ol>
|         </li>
|         <li>Javascript</li>
|     </ol>
| </body>
| </html>
-----+

```

Trong Ví dụ 1-2-7, phần tử đầu trong danh sách cha có số thứ tự là 3 (tức là III trong danh sách dạng I); còn danh sách con của phần tử HTML lại bắt đầu từ thứ tự thứ 2 (tức là ký tự b trong danh sách dạng a); cuối cùng là danh sách con của phần tử CSS do không chỉ định

thứ tự bắt đầu nên sẽ có thứ tự bắt đầu mặc định là 1 (tức là số 1 trong danh sách dạng 1).



**Hình 1-2-7: Trang Vidu1-2-7.html, một ví dụ về danh sách có thứ tự**

#### **1.2.4 Thẻ chèn hình ảnh, âm thanh**

Hình ảnh trên Web được phân làm hai dạng. Dạng thứ nhất là hình ảnh làm nền, tức là chúng ta có thể viết chữ lên hình. Ở dạng này, chúng ta có thẻ chèn hình trong thuộc tính *background* của một số thẻ hỗ trợ ảnh nền như *<body>*, *<table>*, *<td>*, .... Dạng thứ hai, hình ảnh được xem như một đối tượng trên Web, nó chiếm một vị trí trên Web như một nội dung của trang. Trong trường hợp này, chúng ta dùng thẻ *<img>*, đây là một thẻ HTML đơn với cú pháp như sau:

```

```

Trong đó:

URL: là đường dẫn đến tập tin hình ảnh cần đưa lên Web.

*Left / right / top / middle / bottom*: canh lề cho ảnh; *absbottom / absmiddle / texttop*: canh lề cho văn bản xung quanh hình.

Chuỗi trong *title* sẽ hiển thị khi đưa con trỏ chuột vào hình.

Số/phần trăm trong *width* và *height* chỉ định độ lớn của ảnh tương ứng theo chiều rộng và chiều cao. Nếu bỏ qua hai thuộc tính này, trình duyệt sẽ hiển thị hình ảnh với kích thước thật của ảnh.

Chuỗi trong *alt* sẽ được hiển thị thay cho hình khi hình không được hỗ trợ hiển thị lên Web.

Nếu muốn hình ảnh có đường viền, bạn nhập n là một số nguyên dương. Mặc định n là 0, nghĩa là ảnh không có đường viền.

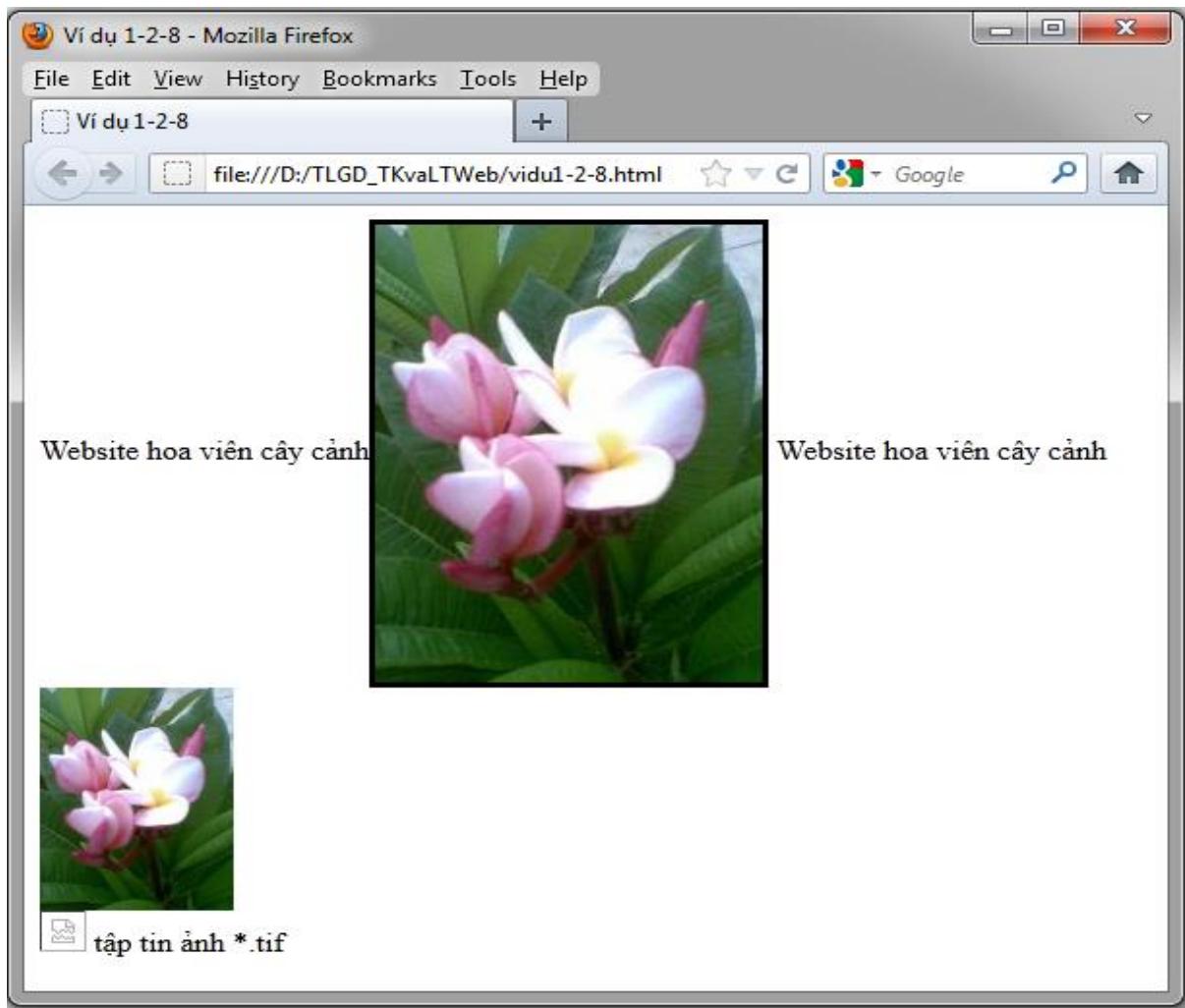
Ví dụ 1-2-8: Thẻ chèn hình ảnh

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Ví dụ 1-2-8</title>
</head>
<body>
    Website hoa viên cây cảnh
    
    Website hoa viên cây cảnh
    <br>
    
    
</body>
</html>
```

Thẻ *<img>* chèn được các tập tin ảnh dạng \*.jpg, \*.jpeg, \*.gif, \*.png.

Trong ví dụ 1-2-8, lần chèn ảnh thứ 3, ảnh có phần mở rộng là .tif không được hỗ trợ hiển thị nên dòng chữ trong thuộc tính *alt* sẽ được hiển thị thay thế ảnh được gọi trong *src*. Trong trường hợp bạn gọi một hình ảnh không được hỗ trợ hiển thị bởi trình duyệt, nhưng

không sử dụng thuộc tính *alt* thì tại vị trí gọi hình, sẽ hiển thị dạng file bị lỗi: .



**Hình 1-2-8: Trang Vidu1-2-8.html**

Âm thanh trên Web cũng được phân làm hai loại: âm thanh nền cho Web và âm thanh được mở trên Web như một trình hát nhạc.

Đối với âm thanh nền hay còn gọi là nhạc nền, khi trang Web vừa được tải lên sẽ xuất hiện âm thanh chạy ngầm bên trong. Bạn cần khai báo thẻ này trong phần `<head>` của trang Web với tên thẻ là `<bgsound>`, đây là một thẻ đơn. Tuy nhiên không phải trình duyệt nào cũng hỗ trợ loại thẻ HTML này.

```
<bgsound src="URL" loop="n">
```

Trong đó:

URL là đường dẫn đến tập tin làm âm thanh nền cho Web, file âm thanh có đuôi:

\*.mepg, \*.avi, \*.mov, \*.au, \*.mid, \*.mp3.

n là số lần lặp lại của file âm thanh, n = -1 nếu muốn lặp lại vô tận.

Trường hợp bạn muốn mở tập tin âm thanh trên Web với một trình hát nhạc, bạn có thể sử dụng cặp thẻ `<embed> ... </embed>`.

```
<embed src="URL" autostart="true/false" loop="true/false"
hidden="true/false" width="n" height="m"></embed>
```

Trong đó:

URL là đường dẫn đến tập tin âm thanh.

*Autostart*: tùy chỉnh chế độ phát tự động (*true*) hay chờ nhấn nút play (*false*).

*Loop*: thiết lập chế độ tự động phát lại (*true*) hay không phát lại (*false*).

*Hidden*: cho phép ẩn chương trình hát nhạc (*true*) hay hiển thị trình hát nhạc trên Web (*false*), mặc định *hidden=“false”*. Đối với một số trình duyệt không hỗ trợ thẻ `<bgsound>` bạn có thể cho phát nhạc nền bằng cách cho chế độ *hidden=“true”*.

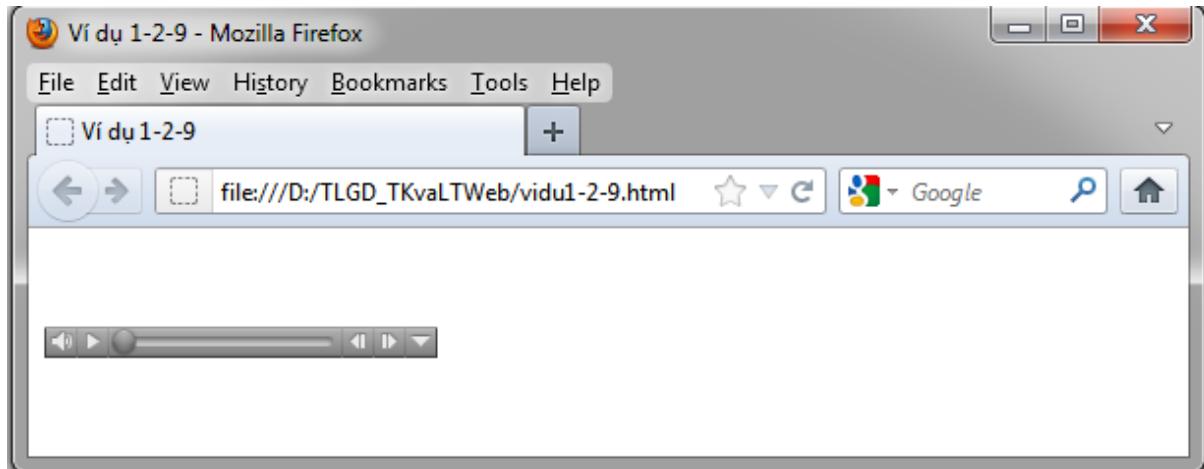
*Width* và *height* là hai thuộc tính chỉ định độ rộng và chiều cao của chương trình hát nhạc trên Web.

Ngoài ra, nếu bạn muốn chèn các tập tin flash hoặc phim ảnh lên Web, thẻ `<embed>` vẫn có thể thực hiện tốt. Tuy nhiên, do các tập tin này có những thông số đặc biệt và tùy theo ý định hiển thị, bạn cần kết hợp thêm cặp thẻ `<object> ... </object>` để được hỗ trợ thêm nhiều thuộc tính hiển thị hơn.

Ví dụ 1-2-9: chèn âm thanh bằng `<bgsound>` và `<embed>`

```
-->
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Ví dụ 1-2-9</title>
<bgsound src="AMTHANH/nhac1.mp3" loop="-1" >
</head>
<body>
    <embed src="AMTHANH/nhac2.mp3" autostart="false"
loop="false" width="200" height="100" ></embed>
</body>
</html>
```

Trình duyệt Internet Explorer hỗ trợ tốt cả `<bgsound>` và `<embed>`. Mozilla Firefox chỉ hỗ trợ `<embed>`.



**Hình 1-2-9: Trang Vidu1-2-9.html**

❖ Câu hỏi (bài tập) củng cố:

1. Hãy thiết kế trang Web có nội dung như sau và lưu bài tập vào D:/hotenSV/ với tên baitap1.html:



2. Hãy thiết kế trang Web có nội dung như sau và lưu bài tập vào D:/hotenSV/ với tên baitap2.html:

**NGÔN NGỮ HTML**  
(Hyper Text Markup Language)

- Cấu trúc trang HTML:

```
<html>
<head>
<title> tiêu đề trang </title>
</head>
<body bgcolor="#445566" text="#aacc88">
    Nội dung hiển thị
</body>
</html>
```
- Các thẻ lệnh:
  - <b> chữ đậm </b>
  - <i> chữ nghiêng </i>
  - <u> chữ gạch chân </u>
- Liên kết:

```
<a href=URL></a>
```

3. Hãy thiết kế trang Web có nội dung như sau và lưu bài tập vào D:/hotenSV/ với tên baitap3.html:

## VÍ DỤ VỀ CÁC THẺ DANH SÁCH

---

- Webpage: trang thông tin hiển thị dưới dạng siêu văn bản được tạo từ ngôn ngữ HTML
- Homepage: trang Web đầu tiên của một Website – cung cấp một cái nhìn tổng quan về Website
- Website: tập hợp những trang Web được liên kết với nhau theo một cấu trúc nào đó của một tổ chức hay cá nhân

**CÁC GIAI ĐOẠN CHUẨN BỊ WEBSITE**

- Giai đoạn thiết kế
- Giai đoạn phát triển

**GIAI ĐOẠN THIẾT KẾ**

- a. Tiêu đề mô tả tài liệu
- b. Tựa đề chính
- c. Đề mục phụ
- d. Mục đích của trang
- e. Mô tả nội dung
- f. Các ảnh được sử dụng
- g. Mô tả về các liên kết

**GIAI ĐOẠN PHÁT TRIỂN**

1. Hình thức và nội dung của trang Web
2. Sự điều hướng trong Website: lược đồ để dẫn đường người xem đi trong Website

4. Hãy thiết kế trang Web có nội dung như sau và lưu bài tập vào D:/hotenSV/ với tên baitap4.html:

## Thực phẩm và Vitamin

- I. Vitamin A
  - 1. Đu đủ
  - 2. Bí đỏ
  - 3. Gan động vật
- II. Vitamin C
  - 1. Cam
  - 2. Cà chua
  - 3. Chanh

5. Bài tập nâng cao 1: Vận dụng những kiến thức đã học ở bài 2, sinh viên hãy thiết kế một trang Web giới thiệu thông tin cá nhân của sinh viên và giới thiệu cho bạn bè trong lớp cùng tham khảo.
6. Bài tập nâng cao 2: Vận dụng những kiến thức đã học ở bài 2, sinh viên hãy thiết kế một website giới thiệu sản phẩm (sinh viên có thể chọn các đề tài giới thiệu sách, quần áo thời trang, điện thoại, máy tính, ...)

## BÀI 3

### **SIÊU LIÊN KẾT VÀ KHUNG**

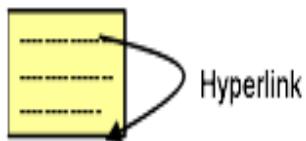
❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Thực hiện tạo Hyperlink và Frame

#### **1.3.1 Siêu liên kết – Hyperlink**

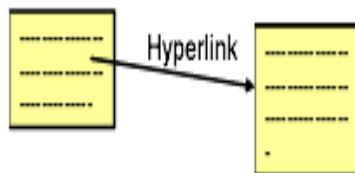
Siêu liên kết – Hyperlink – là mối liên kết giữa hai phần tử thông tin trong một siêu văn bản. Có 3 loại siêu liên kết:

Liên kết trong (Internal link): liên kết các phần tử thông tin trong cùng một tài liệu.



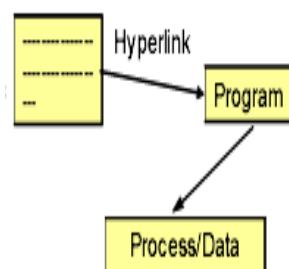
**Hình 1-3-1: Liên kết trong**

Liên kết ngoài (External link): liên kết đến một tài liệu khác, tài liệu được liên kết tới có thể nằm trong cùng một Website hoặc liên kết ra Website khác.



**Hình 1-3-2: Liên kết ngoài**

Liên kết có thể thực thi được (Executable): liên kết gọi thực thi một chương trình hoặc một đoạn mã lệnh. Liên kết này cho phép truy xuất đến cơ sở dữ liệu.



**Hình 1-3-3: Liên kết có thể thực thi được**

### 1.3.1.1 Liên kết trong

Khi nội dung trang Web quá dài, trên trình duyệt sẽ xuất hiện thanh trượt, việc kéo thanh trượt đôi khi làm mất thời gian của người duyệt Web. Khi đó ta cần đặt các liên kết để khi cần người dùng có thể click chọn phần nội dung họ quan tâm mà không cần phải kéo thanh trượt để tìm kiếm đoạn đó.

Để thực hiện điều này, trước tiên cần định nghĩa vị trí đích – nơi sẽ được liên kết đến, ta sử dụng thẻ `<a> ... </a>` theo cú pháp:

```
<a name="TenViTri"> Chuỗi xác định vị trí đích </a>
```

Khi đã có các vị trí đích, bạn chỉ cần tạo liên kết để người dùng click vào theo cú pháp:

```
<a href="#TenViTri"> Từ hiển thị liên kết </a>
```

Để hiểu rõ hơn về cách hoạt động của liên kết trong, mời các bạn tham khảo ví dụ bên dưới

The screenshot shows a Mozilla Firefox window with the title bar 'Ví dụ 1-3-1 - Mozilla Firefox'. The menu bar includes File, Edit, View, History, Bookmarks, Tools, and Help. The address bar shows the URL 'file:///D:/TLGD\_TKvalTWeb/vidu1-3-1.html'. The main content area displays the text 'Tìm hiểu về siêu liên kết' in pink. Below it, there is a paragraph about hyperlinks and three points labeled a, b, and c. A small diagram of a document with an arrow pointing out from it is labeled 'Hyperlink'. At the bottom, there is a note about external links.

Siêu liên kết – Hyperlink – là mối liên kết giữa hai phần tử thông tin trong một siêu văn bản. Có 3 loại siêu liên kết:

- [Liên kết trong \(Internal link\)](#)
- [Liên kết ngoài \(External link\)](#)
- [Liên kết có thể thực thi được \(Executable\)](#)

Trong đó

Liên kết trong (Internal link): liên kết các phần tử thông tin trong cùng một tài liệu.

Hình ảnh minh họa liên kết trong:

Liên kết ngoài (External link): liên kết đến một tài liệu khác, tài liệu được liên kết tới có thể nằm trong cùng một Website hoặc liên kết ra Website khác.

**Hình 1-3-4: Trang Vidu1-3-1.html**

### Ví dụ 1-3-1: Liên kết trong

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Ví dụ 1-3-1</title>
</head>
<body>
    <p align="center"><font color="violet" size="6"><a
name="dautrang">Tìm hiểu về siêu liên kết</a></font></p>
    <p align="justify">
        Siêu liên kết - Hyperlink - là mỗi liên kết giữa hai phần
tử thông tin trong một siêu văn bản. Có 3 loại siêu liên
kết:</p>
    <ol type="a">
        <li><a href="#IL">Liên kết trong (Internal
link)</a></li>
        <li><a href="#EL">Liên kết ngoài (External
link)</a></li>
        <li><a href="#EX">Liên kết có thể thực thi được
(Executable)</a></li>
    </ol>
    Trong đó
    <p align="justify"><a name="IL">Liên kết trong (Internal
link) :</a> liên kết các phần tử thông tin trong cùng một tài
liệu.</p>
    Hình ảnh minh họa liên kết trong: <br>
    
    <p align="justify"><a name="EL">Liên kết ngoài (External
link) :</a> liên kết đến một tài liệu khác, tài liệu được liên
kết tới có thể nằm trong cùng một Website hoặc liên kết ra
Website khác.</p>
    Hình ảnh minh họa liên kết ngoài: <br>
    
    <p align="justify"><a name="EX">Liên kết có thể thực thi
được (Executable) : </a>liên kết gọi thực thi một chương trình
hoặc một đoạn mã lệnh. Liên kết này cho phép truy xuất đến cơ
sở dữ liệu.</p>
    Hình ảnh minh họa liên kết có thể thực thi được: <br>
    
    <a href="#dautrang">Về đầu trang</a>
</body>
</html>
```

Ở ví dụ 1-3-1 này, vì nội dung trang quá dài sẽ gây khó khăn trong việc tìm kiếm nội dung cần quan tâm của người đọc nên chúng tôi vận dụng thẻ liên kết trong như sau: đoạn mã

lệnh `<a name="IL">Liên kết trong (Internal link)</a>` đánh dấu vị trí đích sẽ được trỏ đến khi người dùng click chuột vào liên kết Liên kết trong. Như vậy để tạo liên kết chúng tôi dùng đoạn mã sau: `<a href="#IL">Liên kết trong (Internal link)</a>`. Trong đó, thuộc tính `href` chỉ định vị trí cần trỏ đến là `IL` được đánh dấu trong thuộc tính `name` của thẻ `<a>`. Tương tự, `<a href="#dautrang">Về đầu trang</a>` có `href` đang chỉ về vị trí của một thẻ `<a>` ở phần đầu trang Web `<a name="dautrang">Tìm hiểu về siêu liên kết</a>`. Do đó, nếu ở vị trí cuối trang người duyệt Web click vào liên kết Về đầu trang sẽ tự động trượt về đầu trang.

### 1.3.1.2 Liên kết ngoài

Đối với liên kết ngoài, chúng ta chỉ dùng một cú pháp duy nhất là:

```
<a href="URL" target="_blank/_new/_parent/_seft/_top/frame-name"> Từ hiển thị liên kết </a>
```

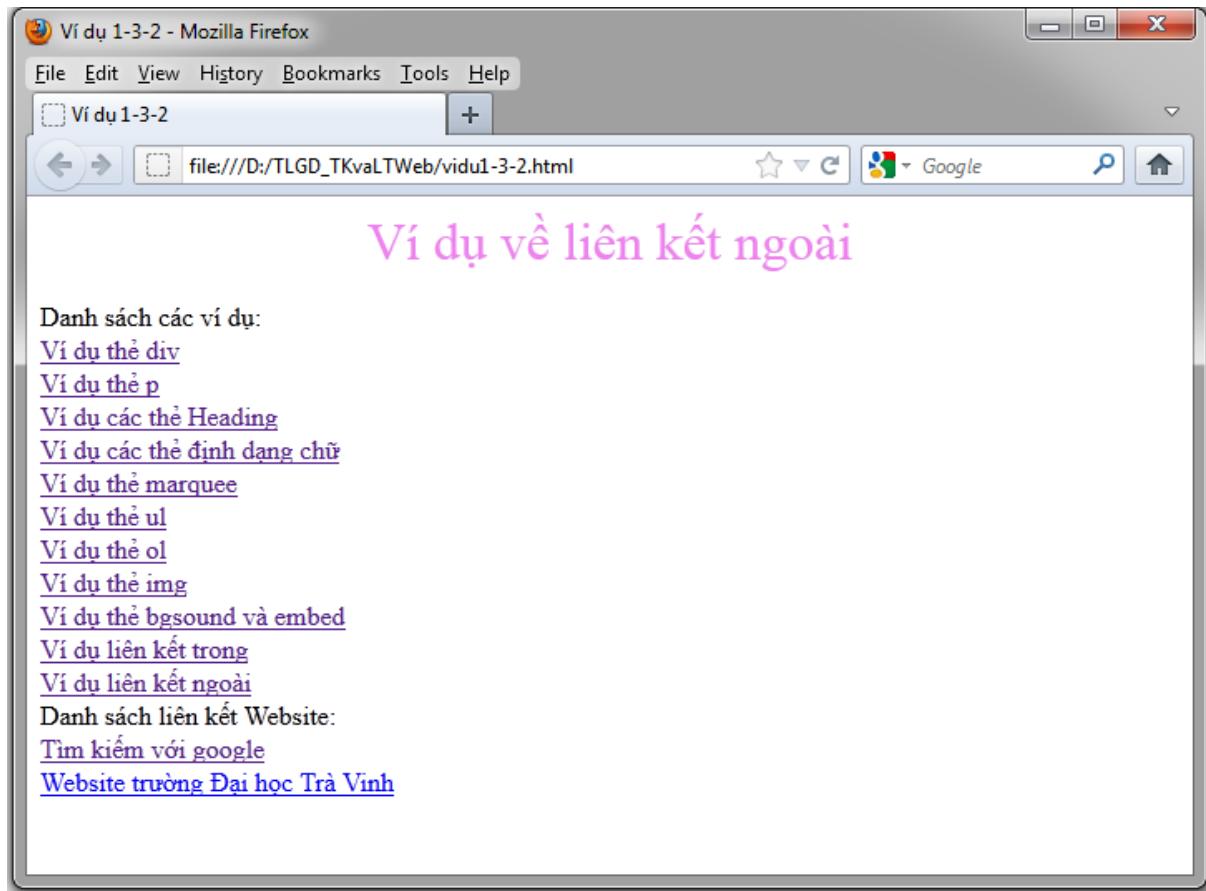
Trong đó:

URL có thể là địa chỉ tương đối hoặc tuyệt đối, có thể là đường dẫn đến một tập tin HTML khác trong cùng một Website hay địa chỉ của một Website khác.

*Target* là thuộc tính quy định cách mở liên kết. Mặc định là mở trên chính trang hiện hành `_parent / _seft`. Nếu bạn muốn mở liên kết ở một cửa sổ mới hoặc ở một Tab mới trên trình duyệt, bạn có thể chỉ định giá trị `_new / _blank` cho thuộc tính `target`. Nếu bạn muốn hiển thị nội dung sắp được liên kết đến trong một khung nào đó (định nghĩa khung sẽ được nói rõ ở phần sau) thì hãy đặt `target` chính bằng tên khung. Phần này chúng tôi sẽ minh họa ở phần Khung – frame.

### Ví dụ 1-3-2: Liên kết ngoài

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Ví dụ 1-3-2</title>
</head>
<body>
<p align="center"><font color="violet" size="6">Ví dụ về
liên kết ngoài</font></p>
Danh sách các ví dụ:<br>
<a href="Vidul-2-1.html">Ví dụ thẻ div</a> <br>
<a href="vidul-2-2.html" target="_self">Ví dụ thẻ
p</a><br>
<a href="vidul-2-3.html" target="_blank" >Ví dụ các thẻ
Heading</a><br>
<a href="vidul-2-4.html">Ví dụ các thẻ định dạng
chữ</a><br>
<a href="vidul-2-5.html">Ví dụ thẻ marquee</a><br>
<a href="vidul-2-6.html">Ví dụ thẻ ul</a><br>
<a href="vidul-2-7.html">Ví dụ thẻ ol</a><br>
<a href="vidul-2-8.html">Ví dụ thẻ img</a><br>
<a href="vidul-2-9.html">Ví dụ thẻ bgsound và
embed</a><br>
<a href="vidul-3-1.html">Ví dụ liên kết trong</a><br>
<a href="vidul-3-2.html">Ví dụ liên kết ngoài</a><br>
Danh sách liên kết Website:<br>
<a href="http://google.com">Tìm kiếm với google</a><br>
<a href="http://tvu.edu.vn">Website trường Đại học Trà
Vinh</a>
</body>
</html>
```



Hình 1-3-5: Trang Vidu1-3-2.html

### 1.3.2 Khung – frame

#### 1.3.2.1 Thẻ `<frameset> ... </frameset>`

Khi cần hiển thị trên trình duyệt nhiều nội dung, chủ đề khác nhau trên cùng một trang thì một giải pháp có thể đáp ứng cho trường hợp này là khung – frame. Chúng ta có thể chia một trang thành nhiều khung, mỗi khung chứa thông tin của một trang HTML. Mỗi khung sẽ có một thanh cuộn riêng khác với thanh cuộn của trình duyệt Web đã cung cấp cho trang Web chính.

Trong các trang Web thông thường, phần thân của trang được đặt giữa cặp thẻ `<body> ... </body>`. Trong trường hợp dùng khung, cặp thẻ `<body> ... </body>` sẽ được thay bằng cặp thẻ `<frameset> ... </frameset>`.

Khung là một lựa chọn đơn giản để có được một giao diện Website thống nhất và ổn định. Tuy nhiên, do mỗi khung sẽ có thanh trượt riêng khi nội dung không đủ hiển thị trong khung, nên đôi khi sử dụng khung để làm giao diện chính cho trang đôi khi gây ra sự rườm rà, mất thẩm mĩ.

Có hai cách chia khung chuẩn: chia theo chiều ngang của trang Web chính, hoặc chia theo chiều dọc của trang Web chính. Từ hai kiểu chia này, người ta có thể kết hợp lại để có được giao diện phức tạp hơn.

Cách chia khung theo chiều ngang có cú pháp như sau:

```
<html>
<head><title>Nội dung tiêu đề </ title ></head>
<frameset rows="a, b,...,n" >
    <frame name="Name_1" Src="URL_1">
    <frame name="Name_2" Src="URL_2">
    ...
    <frame name="Name_n" Src="URL_n">
</frameset>
</html>
```

Trong đó:

Trang gồm có bao nhiêu khung sẽ do bạn chỉ định trong thuộc tính *rows* của *<frameset>*, với chiều cao của mỗi khung được xác định bằng số pixel hoặc bằng phần trăm.

Tên khung được xác định trong thuộc tính *name*, tên khung là một thuộc tính quan trọng để có thể chỉ định liên kết ngoài từ trang này xuất hiện nội dung tương ứng ở khung kia.

Mỗi khung được định nghĩa bằng một *<frame>* nằm trong cặp thẻ *<frameset> ... </frameset>*. Tại vị trí gọi *<frame>*, bạn muốn hiển thị tập tin nào thì gọi đường dẫn đến file đó trong thuộc tính *src*.

Tương tự, cách chia khung theo chiều dọc có cú pháp như sau:

```
<html>
<head><title>Nội dung tiêu đề </ title ></head>
<frameset cols="a, b,...,n" >
    <frame name="Name_1" Src="URL_1">
    <frame name="Name_2" Src="URL_2">
    ...
    <frame name="Name_n" Src="URL_n">
</frameset>
</html>
```

Trong đó:

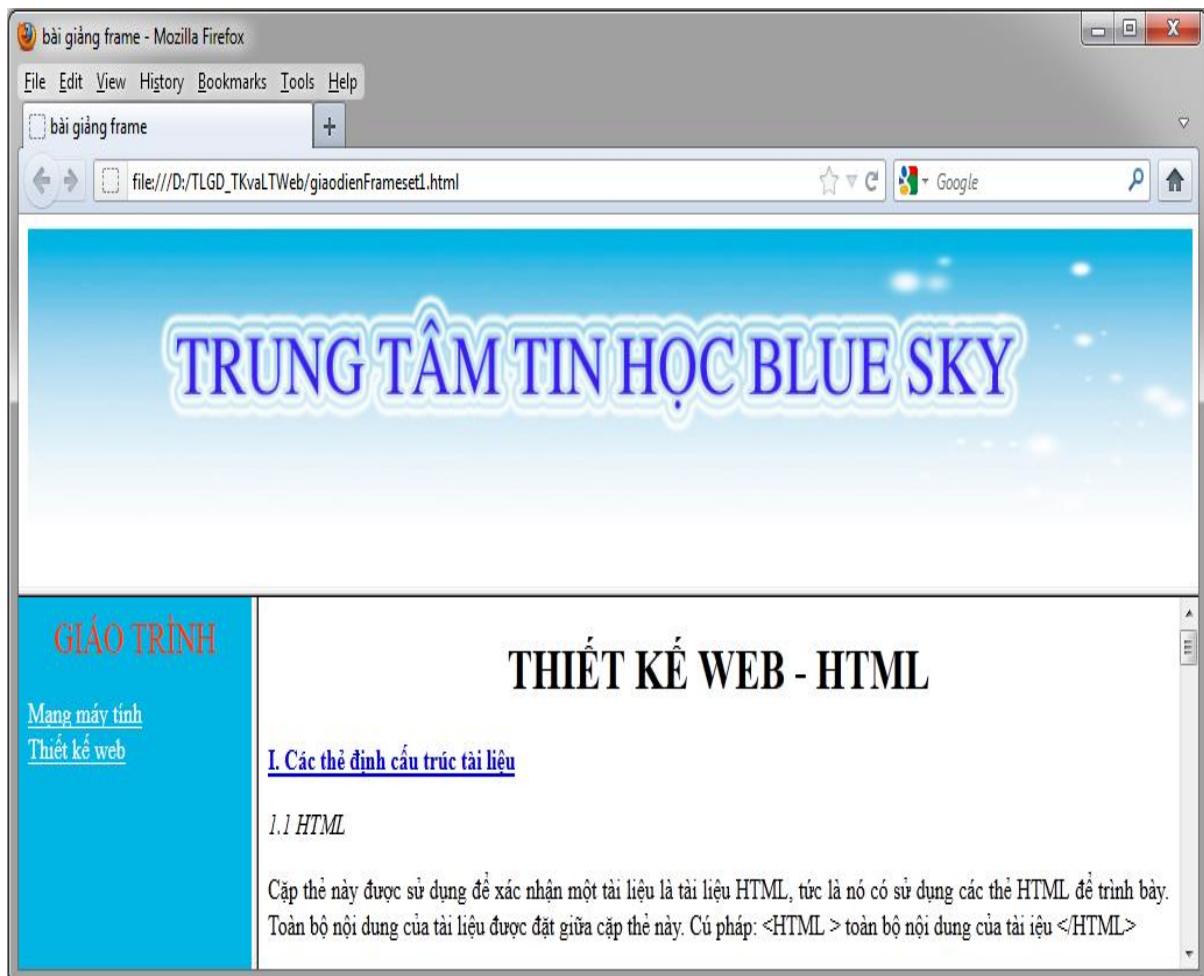
Trang gồm có bao nhiêu khung sẽ do bạn chỉ định trong thuộc tính *cols* của *<frameset>*, với chiều rộng của mỗi khung được xác định bằng số pixel hoặc bằng phần

trăm.

Tên khung được xác định trong thuộc tính *name*, tên khung là một thuộc tính quan trọng để có thể chỉ định liên kết ngoài từ trang này xuất hiện nội dung tương ứng ở khung kia.

Mỗi khung được định nghĩa bằng một *<frame>* nằm trong cặp thẻ *<frameset>* ... *</frameset>*. Tại vị trí gọi *<frame>*, bạn muốn hiển thị tập tin nào thì gọi đường dẫn đến tập tin đó trong thuộc tính *src*.

Giả sử, để tạo được giao diện chính cho Website như hình bên dưới:



**Hình 1-3-6: giaodienFrameset1.html**

Chúng ta cần tạo các trang riêng rẽ như sau: trang banner, trang chứa các liên kết và các trang nội dung chính, sau đó đưa các trang này vào các khung của trang **giaodienFrameset1.html**.



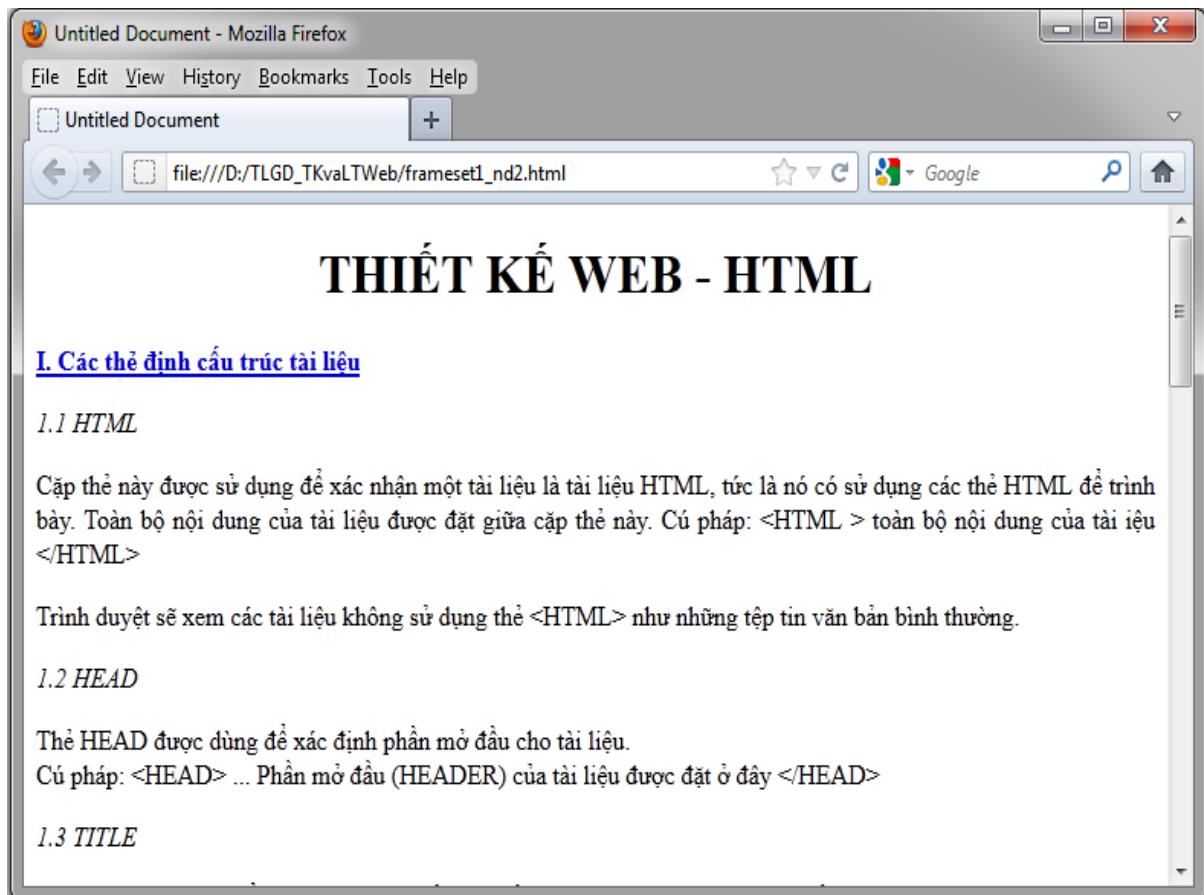
Hình 1-3-7: Trang frameset1\_banner.html



Hình 1-3-8: Trang frameset1\_menu.html



Hình 1-3-9: Trang frameset1\_nd1.html



Hình 1-3-10: Trang frameset1\_nd2.html

Nhìn vào hình 1-3-6 ta thấy, đây là một frameset dạng tổng hợp. Frameset lớn có 2 dòng. Trong đó, dòng 1 chứa trang **frameset1\_banner.html**; dòng 2 là một frameset nhỏ có dạng cột gồm cột bên trái chứa trang **frameset1\_menu.html** và cột lớn bên phải đang chứa trang **frameset1\_nd2.html**. Bạn cần phải biết rằng trên trang giao diện chính này, nếu người dùng click vào liên kết nào trong menu bên trái thì đều hiển thị trang nội dung tương ứng ở khung thứ 2 của frameset nhỏ. Để làm được điều đó, chúng ta cần xác định tên của khung chứa nội dung, giả sử ta đặt *name=“body”* cho khung này. Sau đó thêm thuộc tính *target=“body”* cho các thẻ *<a>* trong trang **frameset1\_menu.html** để cùng hiển thị nội dung trong khung có tên là *body*. Hãy xem mã nguồn của trang **giaodienFrameset1.html**.

Ví dụ 1-3-3: Mã nguồn trang **giaodienFrameset1.html**

```
--> <html>
| <head>
|   <meta http-equiv="Content-Type" content="text/html;
Tài liệu giảng dạy Môn Thiết kế và Lập trình Web
|   <title>bài giảng frame</title>
| </head>
| <frameset rows="40%, *">
|   <frame name="banner" src="frameset1_banner.html" />
```

Trong ví dụ 1-3-3 này, bạn cần chú ý vào thuộc tính `name="body"` của thẻ `<frame>` thứ 3 vì chúng tôi có gọi lại trong ví dụ 1-3-4 bên dưới.

Ví dụ 1-3-4: Mã nguồn trang frameset1\_menu.html

```
-----  
|<html>  
|<head>  
|<meta http-equiv="Content-Type" content="text/html;  
| charset=utf-8" />  
|<title>menu trái</title>  
|</head>  
|<body bgcolor="#00B5E4">  
|    <p align="center"><font color="#FF3300" size="5">GIÁO  
| TRÌNH</font></p>  
|    <a href="frameset1_nd1.html" target="body">  
|        <font color="#FFFFFF">Mạng máy tính</font>  
|    </a>  
|    <br />  
|    <a href="frameset1_nd2.html" target="body">  
|        <font color="#FFFFFF">Thiết kế web</font>  
|    </a>  
|</body>  
</html>  
-----
```

Ở ví dụ 1-3-4 này, chúng tôi xin giải thích thêm về thuộc tính `target` trong thẻ `<a>`. Để nội dung các trang **frameset1\_nd1.html** và **frameset1\_nd2.html** hiển thị được vào cột thứ 2 của dòng thứ 2 trong giao diện trang **giaodienFrameset1.html**, tức là `<frame>` thứ 3 trong mã nguồn của ví dụ 1-3-3, chúng tôi đặt `target="body"` với `body` là tên của frame nằm ở vị trí cột 2 dòng 2 trên giao diện.

### 1.3.2.2 *Thẻ <noframe> ... </noframe>*

Phần tử *<noframe>* được sử dụng để chỉ nội dung thay thế cho frame khi trình duyệt không hỗ trợ frame.

Cú pháp:

```
<html>
<head>
<title>Nội dung tiêu đề </ title ></head>
<FRAMESET>
    <!-- Các định nghĩa khung -->
</FRAMESET>
<NOFRAME>
    <BODY>
        <!--Giao diện thay thế -->
    </BODY>
</NOFRAME>
</html>
```

### 1.3.2.3 *Thẻ <iframe> ... </iframe>*

Nếu muốn hiển thị văn bản chung với khung chứa nội dung của trang khác thì phải tạo một khung bên trong trang bằng thẻ *<iframe>*, khi trình duyệt không hỗ trợ thì nội dung trong IFRAME sẽ không thể hiển thị được.

Tại vị trí muốn chèn frame, nhập vào cú pháp:

```
<iframe src="URL" name="ten_iframe" width="n" height="m"
align="left/ right/ middle/ top/ bottom">
    Nội dung thay thế khi trình duyệt không chấp nhận khung
</iframe>
```

Trong đó:

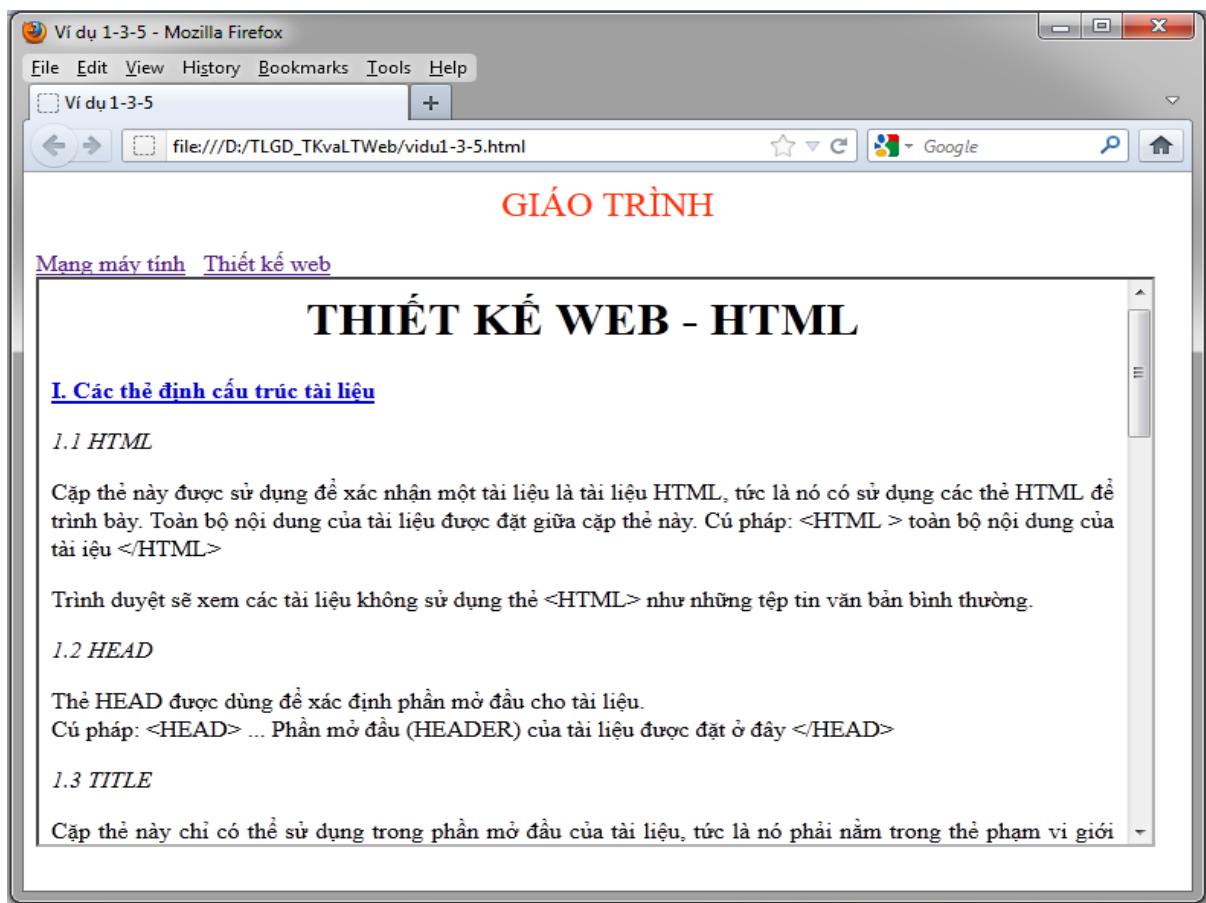
URL là đường dẫn đến trang muốn chèn nội dung vào trang hiện hành.

Bạn cần chỉ định chiều rộng và chiều cao của khung iframe trên giao diện Web.

Có thẻ canh lè cho khung bằng thuộc tính *align*.

Ví dụ 1-3-5: Kết hợp *<iframe>* vào trang Web

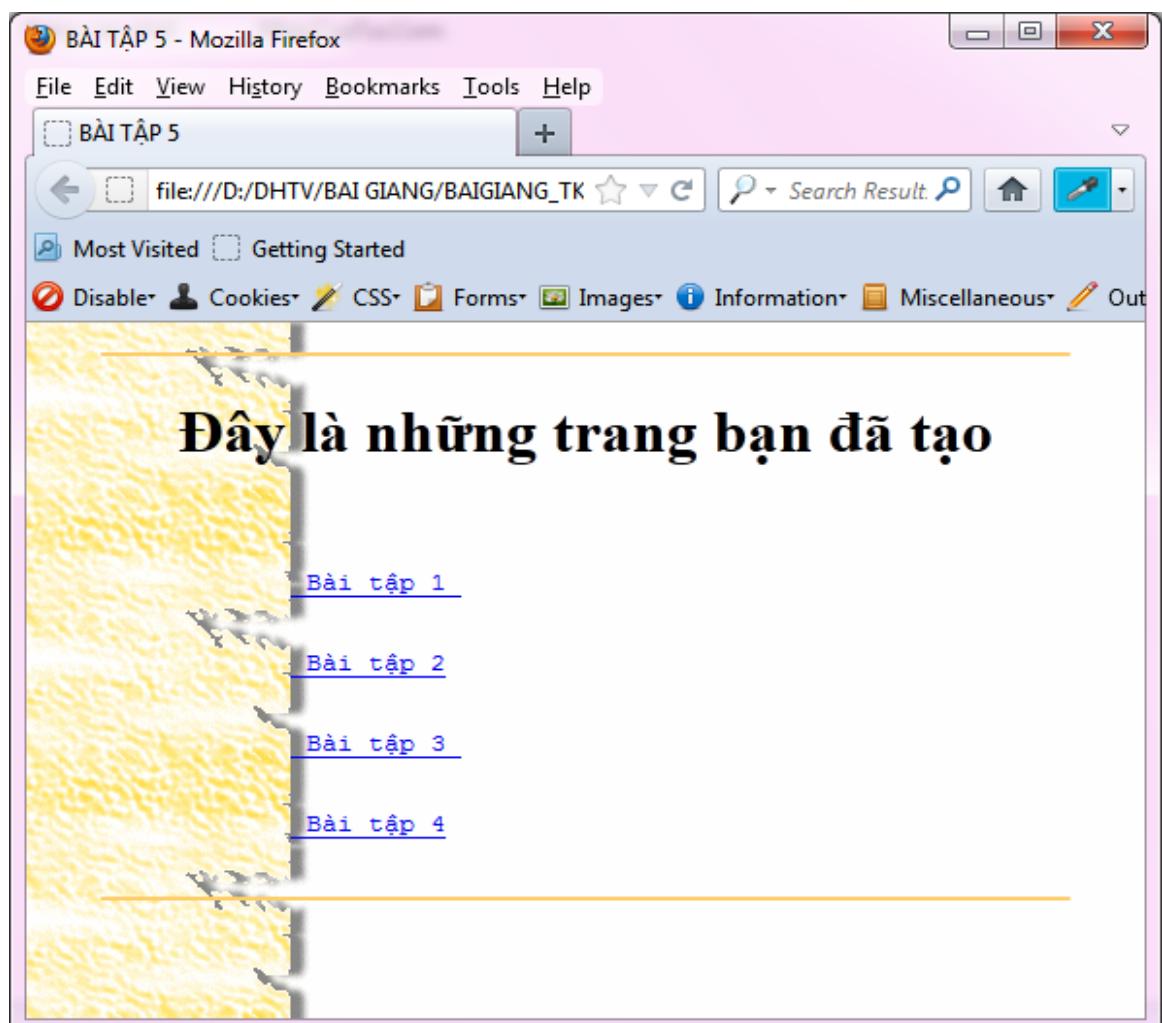
```
----->
| <html>
| <head>
|   <meta http-equiv="Content-Type" content="text/html;
Tài liệu giảng dạy Môn Thiết kế và Lập trình Web
|   <title>Ví dụ 1-3-5</title>
| </head>
| <body>
|   <p align="center"><font color="#FF3300" size="5">GIÁO
```



Hình 1-3-11: Trang Vidu1-3-5.html

❖ Câu hỏi (bài tập) củng cố:

1. Hãy thiết kế trang Web có nội dung như sau và liên kết đến các bài tập đã làm ở bài CÁC THẺ HTML CƠ BẢN. Lưu bài tập vào D:/hotenSV/ với tên baitap5.html:



2. Vận dụng thẻ liên kết trong để thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap6.html:

## VÍ DỤ VỀ LIÊN KẾT TRONG

---

### Danh mục truyện hay

---

[Qua sông](#) [Về đầu trang](#)

Có một người đàn ông đứng bên bờ sông hỏi một cậu bé chăn trâu gần đó:  
- Sông này có sâu không cháu?  
Cậu bé trả lời:  
- Nông lâm bác ạ!  
Người đàn ông lội sang sông, mới lội một đoạn đã ngập đầu. Quay lại, người đàn ông tức giận hỏi cậu bé:  
- Sao cháu bảo sông này nông lắm?  
Cậu bé trả lời:  
- Cháu thấy con vịt chân ngắn thế mà nó cũng lội sang được.

[Thành thật với tình yêu](#) [Về đầu trang](#)

Chiều hoàng hôn, cặp tinh nhân tay trong tay. Cô gái thắc mắc về những tật xấu của người bạn trai mình. Tuy nhiên, nàng hỏi gì, chàng trai cũng khẳng định mình không hề có những tật xấu đó. Nàng lại tiếp:  
- Vậy ngoài em ra, anh còn để ý đến cô nào không?  
Chàng:  
- Em phải tin anh chứ. Anh chỉ có duy nhất em mà thôi!  
Cuối cùng nàng hỏi:  
- Nhưng anh cũng phải có một thói xấu gì đó chứ?  
Chàng thành thật:  
- À, đúng là thỉnh thoảng anh có hay nói dối.

[Những điều vô giá](#) [Về đầu trang](#)

Người mẹ đang bận rộn nấu bữa tối trong bếp, bắt ngò cậu con trai bé bỏng chạy ầm vào, và đưa cho mẹ một mẩu giấy nhỏ. Sau khi lau tay vào chiếc tạp dề, người mẹ mở tờ giấy ra và đọc:

1. Đẹp phòng của con: 1<sup>đ</sup>  
2. Đì chọ cùng với mẹ: 50 xu  
3. Trông em giúp mẹ: 25 xu  
4. Bồ rác: 1<sup>đ</sup>  
5. Kết quả học tập tốt: 5<sup>đ</sup>

=> Mẹ nợ con tổng cộng: 7.75<sup>đ</sup>

Sau khi đọc xong, người mẹ nhìn cậu con trai đang đứng chờ với vẻ mặt đầy hi vọng. Bà cầm bút lên, lật mặt sau của tờ giấy và viết:  
- Chín tháng mười ngày con nằm trong bụng mẹ.  
- Những lúc mẹ bên cạnh chăm sóc, cầu nguyện mỗi khi con ốm đau.  
- Những giọt nước mắt con làm mẹ khóc trong những năm qua.  
- Những đêm mẹ không ngủ vì lo lắng cho tương lai của con.  
- Những đồ chơi, thức ăn, quần áo mà mẹ đã nuôi con trong suốt mấy năm qua.  
- Và đặc hơn cả chính là tình yêu của mẹ dành cho con.

Các bạn thấy đây, Cha Mẹ đã dày công sinh thành và nuôi nấng chúng ta trưởng thành. Cho dù cuộc sống có bao vất vả khó khăn, Cha Mẹ vẫn luôn dành cho con cái những điều tốt đẹp nhất. Thế còn chúng ta, chúng ta đã làm được gì cho Cha Mẹ mình?

[Chiếc hộp tình yêu](#) [Về đầu trang](#)

Một người cha nghèo đã nỗi giận khi đưa con gái 3 tuổi của ông cắt cuộn giấy quý ra thành từng mảnh nhỏ trang trí một cái hộp giấy. Nhưng sáng sớm hôm sau, đứa con gái nhỏ vẫn mang hộp quà đến và tặng cho cha. Người cha cảm thấy bồi rôi vì con giận dữ của mình hôm trước nhưng rồi con giận dữ lại bùng lên khi ông mở ra, thấy cái hộp trống rỗng.

Đứa con gái nhỏ ngược nhìn cha, nuốt nước mắt rung rung: "Bố ơi, đó đâu phải là cái hộp rỗng con đã thổi đáy những nụ hòn vào hộp để tặng bố mà!"

Người cha giật mình. Ông vỗ tay ôm lấy đứa con gái nhỏ và xin con tha thứ.

Sau đây không bao lâu, đứa bé qua đời trong một tai nạn. Nhiều năm sau, người cha vẫn khu khu giữ cái hộp giấy bên mình, mỗi khi gặp chuyện nán lòng, ông lấy ra một nụ hồn tưởng tượng và nghĩ đến tình yêu mà đứa con gái bé bỏng của ông đã thổi vào chiếc hộp.

Trong cuộc sống, đôi khi chúng ta nhận được những món quà quý giá như vậy nhưng lại vô tình bỏ qua.



## Nội dung hỗ trợ cho baitap6.html:

### QUA SÔNG

Có một người đàn ông đứng bên bờ sông hỏi một cậu bé chăn trâu gần đó:

- Sông này có sâu không cháu?

Cậu bé trả lời:

- Nông lâm bác ạ!

Người đàn ông lội sang sông, mới lội một đoạn đã ngập đầu. Quay lại, người đàn ông tức giận hỏi cậu bé:

- Sao cháu bảo sông này nông lâm?

Cậu bé trả lời:

- Cháu thấy con vịt chân ngắn thế mà nó cũng lội sang được.

### THÀNH THẬT VỚI TÌNH YÊU

Chiều hoàng hôn, cặp tình nhân tay trong tay. Cô gái thắc mắc về những tật xấu của người bạn trai mình. Tuy nhiên, nàng hỏi gì, chàng trai cũng khẳng định mình không hề có những tật xấu đó. Nàng lại tiếp:

- Vậy ngoài em ra, anh còn để ý đến cô nào không?

Chàng:

- Em phải tin anh chứ. Anh chỉ có duy nhất em mà thôi!

Cuối cùng nàng hỏi:

- Nhưng anh cũng phải có một thói xấu gì đó chứ?

Chàng thành thật:

- À, đúng là thỉnh thoảng anh có hay nói dối.

### NHỮNG ĐIỀU VÔ GIÁ

Người mẹ đang bận rộn nấu bữa tối trong bếp, bất ngờ cậu con trai bé bỏng chạy vào, và đưa cho mẹ một mẩu giấy nhỏ. Sau khi lau tay vào chiếc tạp dề, người mẹ mở tờ giấy ra và đọc:

1. Dọn dẹp phòng của con: 1\$
2. Đi chợ cùng với mẹ: 50 xu
3. Trông em giúp mẹ: 25 xu
4. Đổ rác: 1\$
5. Kết quả học tập tốt: 5\$

### Mẹ nợ con tổng cộng: 7.75\$

Sau khi đọc xong, người mẹ nhìn cậu con trai đang đứng chờ với vẻ mặt đầy hi vọng. Bà cầm bút lên, lật mặt sau của tờ giấy và viết:

- Chín tháng mười ngày con nằm trong bụng mẹ.
- Những lúc mẹ bên cạnh chăm sóc, cầu nguyện mỗi khi con ôm đau.
- Những giọt nước mắt con làm mẹ khóc trong những năm qua.
- Những đêm mẹ không ngủ vì lo lắng cho tương lai của con.
- Những đồ chơi, thức ăn, quần áo mà mẹ đã nuôi con trong suốt mấy năm qua.
- Và đắt hơn cả chính là tình yêu của mẹ dành cho con.

Các bạn thấy đây, Cha Mẹ đã dày công sinh thành và nuôi nấng chúng ta trưởng thành. Cho dù cuộc sống có bao vất vả khó khăn, Cha Mẹ vẫn luôn dành cho con cái những điều tốt đẹp nhất. Thế còn chúng ta, chúng ta đã làm được gì cho Cha Mẹ mình?

Nội dung hỗ trợ cho baitap6.html (tt):

### CHIẾC HỘP TÌNH YÊU

Một người cha nghèo đã nỗi giận khi đứa con gái 3 tuổi của ông cắt cuộn giấy quý ra thành từng mảnh nhỏ trang trí một cái hộp giấy. Nhưng sáng sớm hôm sau, đứa con gái nhỏ vẫn mang hộp quà đến và tặng cho cha. Người cha cảm thấy bối rối vì cơn giận dữ của mình hôm trước nhưng rồi cơn giận dữ lại bùng lên khi ông mở ra, thấy cái hộp trống rỗng.

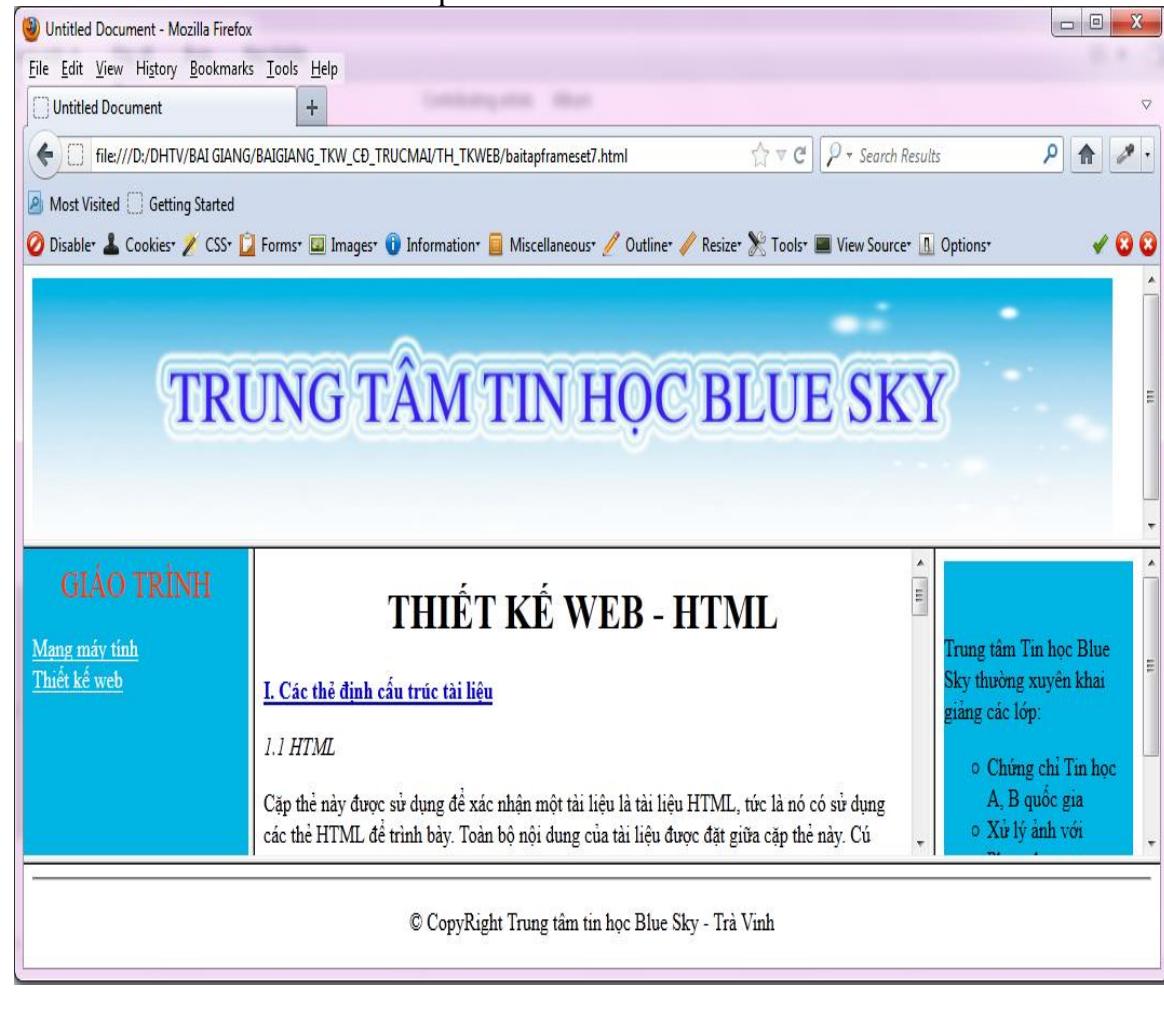
Đứa con gái nhỏ ngược nhìn cha, nước mắt rung rung: "Bố ơi, đó đâu phải là cái hộp rỗng, con đã thổi đầy những nụ hôn vào hộp để tặng bố mà"

Người cha giật mình. Ông vòng tay ôm lấy đứa con gái nhỏ và xin con tha thứ.

Sau đây không bao lâu, đứa bé qua đời trong một tai nạn. Nhiều năm sau, người cha vẫn khư khư giữ cái hộp giấy bên mình, mỗi khi gặp chuyện nản lòng, ông lấy ra một nụ hôn tưởng tượng và nghĩ đến tình yêu mà đứa con gái bé bỏng của ông đã thổi vào chiếc hộp.

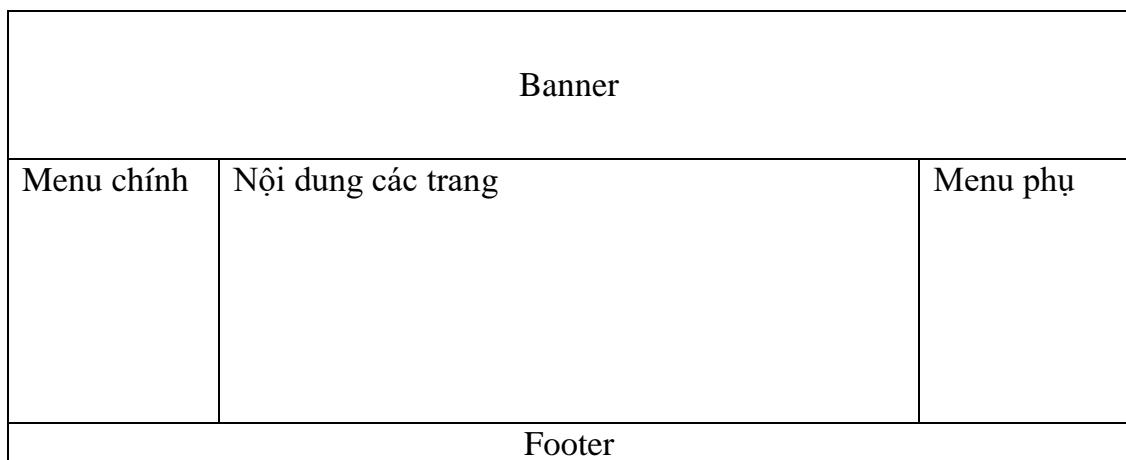
Trong cuộc sống, đôi khi chúng ta nhận được những món quà quý giá như vậy nhưng lại vô tình bỏ qua.

3. Dùng frameset để tạo giao diện cho trang Web sau. Lưu bài tập vào D:/hotenSV/ với tên baitap7.html

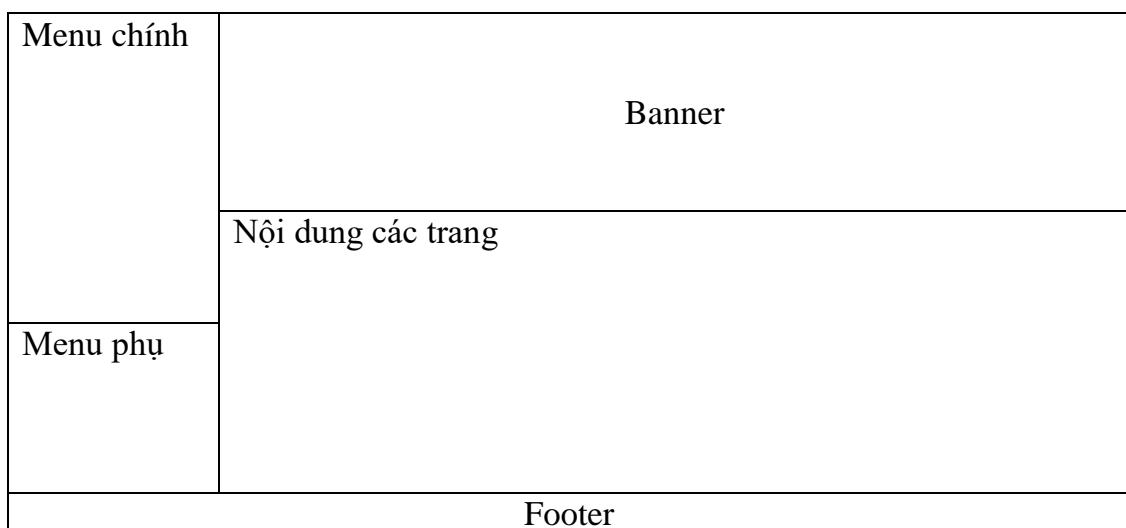


4. Bài tập nâng cao: Hãy dùng frameset để thiết kế giao diện cho bài tập nâng cao 1, 2 ở bài học trước theo các dạng Layout sau:

Layout 1:



Layout 2:



## BÀI 4

### **BẢNG BIỂU TRÊN WEB**

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Thực hiện tạo và định dạng Table trong thiết kế trang Web

#### **1.4.1 Tạo bảng**

Khi bạn muốn thiết kế bảng dữ liệu trên trang Web, bạn cần phải khai báo thẻ `<table>`. Sau khi khai báo thẻ này, bạn phải khai báo các thẻ dòng `<tr>` và thẻ cột `<td>` hoặc `<th>`. Trong đó, `<th>` sẽ xác định nội dung bên trong nó là tiêu đề, tức là nội dung mặc định được in đậm và canh giữa.

Cấu trúc gọi bảng:

```
<table>
<caption> Tiêu đề bảng </caption>
  <tr>
    <th>tiêu đề cột 1</th>
    ...
    <th>tiêu đề cột n </th>
  </tr>
  <tr>
    <td>nội dung ô 11 </td>
    ...
    <td>nội dung ô 1n </td>
  </tr>
  ...
  <tr>
    <td>nội dung ô n1 </td>
    ...
    <td>nội dung ô nn </td>
  </tr>
</table>
```

#### **1.4.2 Các thuộc tính định dạng bảng**

##### **1.4.2.1 Thuộc tính trong thẻ bảng `<table>`**

Khi thiết lập một bảng biểu trên Web, bạn cần chú ý đến các thuộc tính sau đây:

THUỘC TÍNH	Ý NGHĨA
<i>border</i>	Khởi tạo bảng với đường viền (mặc định là không có đường viền, <i>border=“0”</i> ).
<i>cellpadding</i>	Khoảng cách từ nội dung trong ô đến đường viền.
<i>cellspacing</i>	Khoảng cách giữa các ô trong bảng.
<i>align</i>	Canh lè cho bảng ( <i>align=“center / left / right”</i> ).
<i>width</i>	Xác định chiều rộng cho toàn bảng.
<i>height</i>	Xác định chiều cao cho bảng. Tuy nhiên khi thiết kế bảng, bạn chỉ cần quan tâm đến chiều rộng chứ không quan tâm đến chiều cao, bởi vì chiều cao của bảng phụ thuộc vào dữ liệu bạn có.
<i>background</i>	Chèn ảnh nền cho bảng.
<i>bgcolor</i>	Tô màu nền cho bảng.
<i>bordercolor</i>	Tô màu đường viền cho bảng. Lưu ý một số trình duyệt không hỗ trợ hiển thị màu viền.
<i>id</i>	id nhận dạng dùng để phân biệt các bảng dữ liệu trên cùng một trang.

#### 1.4.2.2 Thuộc tính trong thẻ dòng <tr>

Sau đây là một số thuộc tính thường dùng cho dòng:

THUỘC TÍNH	Ý NGHĨA
<i>align</i>	Canh lè dữ liệu theo hướng ngang ( <i>align=“left / center / justify / right”</i> ).
<i>valign</i>	Canh lè dữ liệu theo hướng đứng ( <i>valign=“top / middle / bottom”</i> ).
<i>height</i>	Xác định chiều cao dòng. Lưu ý, đối với dòng chúng ta không quan tâm đến chiều rộng bởi vì chiều rộng của dòng phụ thuộc vào chiều rộng của bảng.

THUỘC TÍNH	Ý NGHĨA
<i>bgcolor</i>	Tô màu nền cho dòng. Lưu ý, trong dòng chỉ có thể tô màu nền chứ không thể gọi ảnh nền cho dòng.
<i>bordercolor</i>	Tô màu đường viền dòng. Lưu ý một số trình duyệt không hỗ trợ hiển thị màu viền.
<i>id</i>	id nhận dạng để phân biệt giữa các dòng trong bảng.

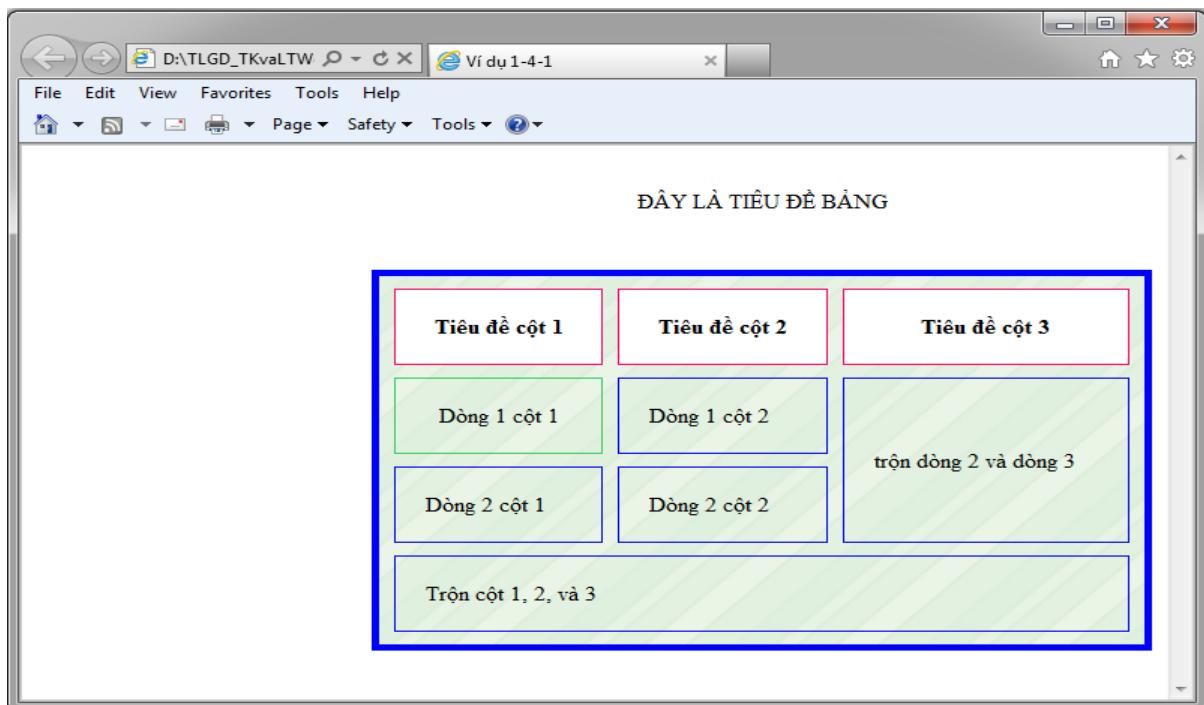
#### 1.4.2.3 Thuộc tính trong các thẻ ô / cột *<th>*, *<td>*

Sau đây là một số thuộc tính thường dùng cho ô / cột:

THUỘC TÍNH	Ý NGHĨA
<i>align</i>	Canh lè dữ liệu trong ô theo hướng ngang. ( <i>align= “left / center / justify / right”</i> ).
<i>valign</i>	Canh lè dữ liệu trong ô theo hướng đứng. ( <i>valign= “top / middle / bottom”</i> ).
<i>background</i>	Gọi ảnh nền cho ô.
<i>bgcolor</i>	Tô màu nền cho ô.
<i>bordercolor</i>	Tô màu đường viền cho ô. Lưu ý một số trình duyệt không hỗ trợ hiển thị màu viền.
<i>colspan</i>	Gộm cột. Ta có <i>colspan= “n”</i> với n là số cột muốn thành một ô.
<i>rowspan</i>	Gộm hàng. Ta có <i>rowspan= “n”</i> với n là số dòng muốn thành một ô.
<i>id</i>	id nhận dạng để phân biệt giữa các ô.
<i>width</i>	Xác định chiều rộng của ô.
<i>height</i>	Xác định chiều cao của ô. Tuy nhiên, đối với ô / cột thì ta thường ít quan tâm đến chiều cao vì chiều cao của ô / cột chính bằng chiều cao của dòng.

### Ví dụ 1-4-1: Tổng hợp các thuộc tính trong bảng

```
|-----|
| <html>
| <head>
| <meta http-equiv="Content-Type" content="text/html;
| charset=utf-8" />
| <title>Ví dụ 1-4-1</title>
| </head>
| <body>
| <table align="right" width="70%" border="5"
| background="hinhanh/ANH_NEN.jpg" bordercolor="blue"
| cellpadding="20" cellspacing="10">
|   <caption> ĐÂY LÀ TIÊU ĐỀ BẢNG </caption>
|   <tr bordercolor="#FF0066" bgcolor="WHITE">
|     <th>Tiêu đề cột 1</th>
|     <th> Tiêu đề cột 2</th>
|     <th> Tiêu đề cột 3</th>
|   </tr>
|   <tr>
|     <td align="center" bordercolor="#33CC66">Dòng 1 cột
|     1</td>
|     <td> Dòng 1 cột 2</td>
|     <td rowspan="2"> trộn dòng 2 và dòng 3 </td>
|   </tr>
|   <tr>
|     <td>Dòng 2 cột 1</td>
|     <td> Dòng 2 cột 2</td>
|   </tr>
|   <tr>
|     <td colspan="3">Trộn cột 1, 2, và 3</td>
|   </tr>
| </table>
| </body>
| </html>
|-----|
```

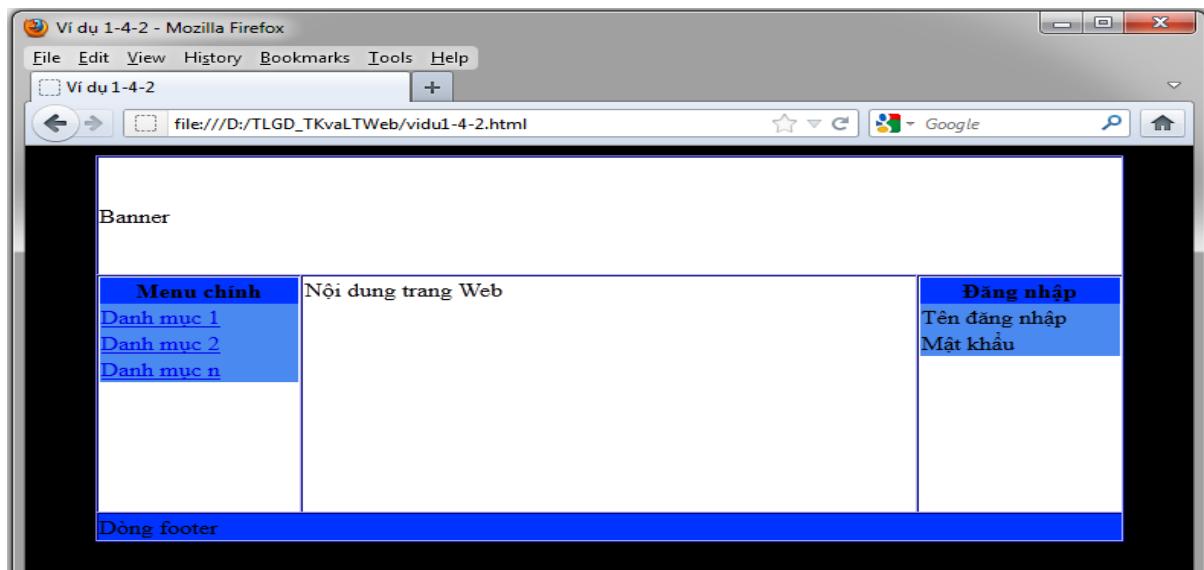


**Hình 1-4-1: Trang Vidu1-4-1.html**

### 1.4.3 Thiết kế giao diện Web bằng `<table>`

Ngoài được dùng để chứa thông tin trên bảng biểu. Thẻ `<table>` còn được vận dụng nhiều trong thiết kế giao diện. Khi giao diện trang có cấu trúc phức tạp, bạn thực hiện khai báo các thẻ `<table>` lồng nhau.

Sau đây là một ví dụ về giao diện sử dụng `<table>`



**Hình 1-4-2: Trang Vidu1-4-2.html**

Ví dụ 1-4-2: Một giao diện bằng *<table>*

```
|-----|<html>
|<head>
|<meta http-equiv="Content-Type" content="text/html;
| charset=utf-8" />
|<title>Ví dụ 1-4-2</title>
|</head>
|<body bgcolor="#000000">
|<table bgcolor="#FFFFFF" border="1" align="center"
| bordercolor="#0000CC" width="90%" cellspacing="0">
|   <tr height="100">
|     <td colspan="3">Banner</td>
|   </tr>
|   <tr>
|     <td valign="top" width="20%">
|       <table bgcolor="#4A89F0" border="0" cellspacing="0"
| width="100%">
|         <tr bgcolor="#0033FF"><th>Menu chính</th></tr>
|           <tr><td><a href="#">Danh mục 1</a></td></tr>
|           <tr><td><a href="#">Danh mục 2</a></td></tr>
|           <tr><td><a href="#">Danh mục n</a></td></tr>
|         </table>
|       </td>
|       <td valign="top">
|         Nội dung trang Web
|       </td>
|       <td valign="top" height="200" width="20%">
|         <table bgcolor="#4A89F0" border="0" cellspacing="0"
| width="100%">
|           <tr bgcolor="#0033FF"><th>Đăng nhập</th></tr>
|             <tr><td>Tên đăng nhập</td></tr>
|             <tr><td>Mật khẩu</td></tr>
|           </table>
|         </td>
|       </tr>
|       <tr>
|         <td colspan="3" bgcolor="#0033FF">
|           Dòng footer
|         </td>
|       </tr>
|     </table>
|   </body>
| </html>
|-----|
```

❖ Câu hỏi (bài tập) củng cố:

- Hãy thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap8.html:

BẢNG ĐƠN GIÁ

Loại hàng	Thương hàng	Đặc biệt	Trung bình
TEA	80.000 đ	60.000 đ	20.000 đ
COFFEE	120.000 đ	100.000 đ	40.000 đ

BẢNG THỐNG KÊ

Tên hàng	Tổng thành tiền	
	Xuất	Nhập
TEA	3.672.000 đ	1.245.120 đ
COFFEE	10.000.000 đ	7.050.000 đ

- Hãy thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap9.html:



BẢNG KÊ TỔNG HỢP

STT	Mã hàng	Tên hàng	Ngày giao	Đơn giá	Số lượng	Thành tiền
1	TNNNS01		05/9/2010			
2	TNNNB05		07/9/2010			
3	CANB03		07/9/2010			
4	PNNB04		08/9/2010			
5	TTNS02		12/9/2010			
6	CANS01		12/9/2010			
7	PNNB03		15/9/2010			
8	TTNB02		17/9/2010			

BẢNG PHỤ

2 ký tự đầu của Mã hàng	Tên Báo	Giá Báo	
		Nhật Báo	Nguyệt San
TT	Tuổi trẻ	1000	2500
TN	Thanh niên	1500	3500
PN	Phụ nữ	2000	5000
CA	Công an	1300	2800

3. Hãy thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap10.html:

The screenshot shows a Firefox browser window with the title "TRANG GIOI THIEU - Mozilla Firefox". The menu bar includes File, Edit, View, History, Bookmarks, Tools, and Help. The toolbar includes Back, Forward, Stop, Refresh, Home, and various developer tools like CSS, Forms, Images, and Information.

The main content area displays four book reviews:

- Tự Học JavaScript**: Javascript là một ngôn ngữ lập trình chính của Web và cũng là ngôn ngữ dễ nắm bắt nhất. Có một số lý do khiến bạn tiếp cận ngôn ngữ này một cách dễ dàng: Javascript được nhúng trong hồ sơ HTML và bạn có thể bắt đầu bằng cách đưa thủ tục nhỏ gọn...
- Vượt Vòng Quanh Nước Hàn Quốc**: Cuốn sách Một Vòng Quanh Nước Hàn Quốc (Du Lịch & Du học) giới thiệu lịch sử, địa lý, kinh tế, văn hóa - xã hội, du lịch, giáo dục, du học của nước Hàn Quốc ...
- Các Phương Pháp Thi Công Xây Dựng**: Các Phương Pháp Thi Công Xây Dựng" là cuốn sách viết cho những người am hiểu về xây dựng, mang tính chất chuyên đề...
- Vượt Qua Thủ Thách**: "Vượt Qua Thủ Thách" giới thiệu đến bạn đọc những câu chuyện như: Chiếc đai lưng màu xanh tía. Một chiếc cầu vòng vẫn không đủ...

Each review includes a small thumbnail image of the book cover.

4. Hãy thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap11.html:

The screenshot shows a Mozilla Firefox browser window with the title "CHỦ ĐỀ DU LỊCH - Mozilla Firefox". The address bar displays the URL: file:///D:/HDTV/BAI GIANG/BAIGIANG\_TKW\_TRUCMAI/TH\_TKWEB/ontap\_chudeDL.html. The browser toolbar includes icons for Disable, Cookies, CSS, Forms, Images, Information, Miscellaneous, Outline, Resizer, Tools, View Source, Options, and three checkmarks.

The main content area features a large blue header "CHỦ ĐỀ DU LỊCH". Below it is a table listing three travel books:

Tên sách	Hình ảnh	Mô tả	Tác giả	Trang	Giá (VND)
Vòng Quanh Nước Hàn Quốc		Cuốn sách Một Vòng Quanh Nước Hàn Quốc (Du Lịch & Du học) giới thiệu lịch sử, địa lý, kinh tế, văn hóa - xã hội, du lịch, giáo dục, du học của nước Hàn Quốc	Trần Vĩnh Bảo	650	72.000
Du Lịch Châu Âu		Cuốn sách "Du Lịch Châu Âu" là một trong những tập nằm trong Bộ sách "Du Lịch Thế Giới". Sách giới thiệu đến bạn đọc một số nước du lịch: Iceland, Na Uy...	Phùng Văn Hòa	412	38.000
Du Lịch Châu Á		Cuốn "Du Lịch Châu Á" là một trong những tập nằm trong Bộ sách "Du Lịch Thế Giới". Nội dung cuốn sách giới thiệu một số nước là điểm du lịch trong khu vực Châu Á: Hàn Quốc, Trung Quốc	Phùng Văn Hòa	620	65.000

5. Hãy thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap12.html:

The screenshot shows a Mozilla Firefox browser window displaying a library catalog page titled "CHỦ ĐỀ KHOA HỌC KỸ THUẬT". The page contains a table with two books listed. The table has columns for Tên sách (Book Name), Hình ảnh (Image), Mô tả (Description), Tác giả (Author), Trang (Pages), and Giá (VNĐ) (Price). The books listed are "Phương Pháp Thi Công Xây Dựng" and "Kỹ Thuật Số".

Tên sách	Hình ảnh	Mô tả	Tác giả	Trang	Giá (VNĐ)
Phương Pháp Thi Công Xây Dựng		Các Phương Pháp Thi Công Xây Dựng" là cuốn sách viết cho những người am hiểu về xây dựng, mang tính chất chuyên đề.	Ngô Văn Quý	500	68.000
Kỹ Thuật Số		Giáo Trình Chuyên Ngành Kỹ Thuật Số bao gồm mọi chủ đề! Bạn sẽ có dịp nghiên cứu về các biểu thức logic cơ bản, mạch logic, mạch tích hợp, flip-flop, thanh ghi, bộ đếm, bộ đếm đa hài.	Nguyễn Trọng Hùng	600	100.000

6. Hãy thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap13.html:

The screenshot shows a Mozilla Firefox window with the title "CHU DE VE TIN HOC - Mozilla Firefox". The address bar displays "file:///D:/DHTV/BAI GIANG/BAIGIANG\_TKW\_TRUCMAI/TH\_TKWEB/ontap\_ch". The main content area shows a table titled "CHỦ ĐỀ TIN HỌC" (Computer Fundamentals). The table has columns for Tên sách (Book Name), Hình ảnh (Image), Mô tả (Description), Tác giả (Author), Trang (Pages), and Giá (VNĐ) (Price). Three rows of data are listed:

Tên sách	Hình ảnh	Mô tả	Tác giả	Trang	Giá (VNĐ)
Tự Học JavaCript		Javascript là một ngôn ngữ lập trình chính của Web và cũng là ngôn ngữ dễ nắm bắt nhất. Có một số lý do khiến bạn tiếp cận ngôn ngữ này một cách dễ dàng: Javascript được nhúng trong hồ sơ HTML	Nguyễn Trường Sinh	384	36.000
Window Xp		Cuốn sách "Windows Xp - Tin Học Văn Phòng" giới thiệu nội dung: Cài Đặt Windows XP Và Những Điều Cơ Bản 7.Các Trình Ứng Dụng Đa Phương Tiện Trong Windows XP.	Phạm Vĩnh Hưng	587	80.000
Các Mẹo Sử Dụng Máy Tính		Cuốn sách gồm các nội dung sau:Sử dụng tập tin (Files), Thiết lập Font tiếng Việt cho hệ thống Windows XP...	Nguyễn Ngọc Tuấn	205	36.000

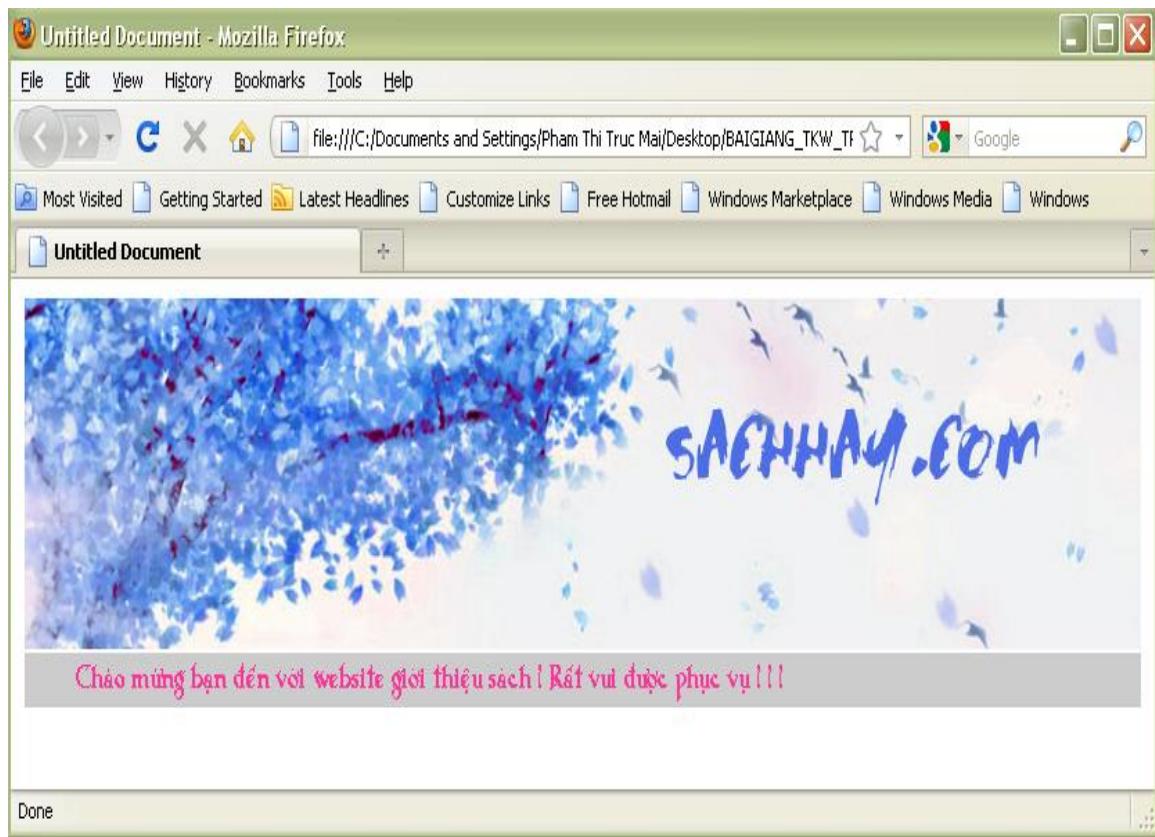
7. Hãy thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap14.html:

The screenshot shows a Mozilla Firefox browser window with the title "CHU DE VE VAN HOA - Mozilla Firefox". The address bar displays "file:///D:/DHTV/BAI GIANG/BAIGIANG\_TKW\_TRUCMAI/TH\_TKWEB/ontap\_ch". The toolbar includes standard buttons like Back, Forward, Stop, Refresh, Home, and Search. Below the toolbar is a menu bar with File, Edit, View, History, Bookmarks, Tools, and Help. A toolbar for "ONTAP CH" is visible, containing icons for Disable, Cookies, CSS, Forms, Images, Information, Miscellaneous, Outliner, Resizer, Tools, and View Source.

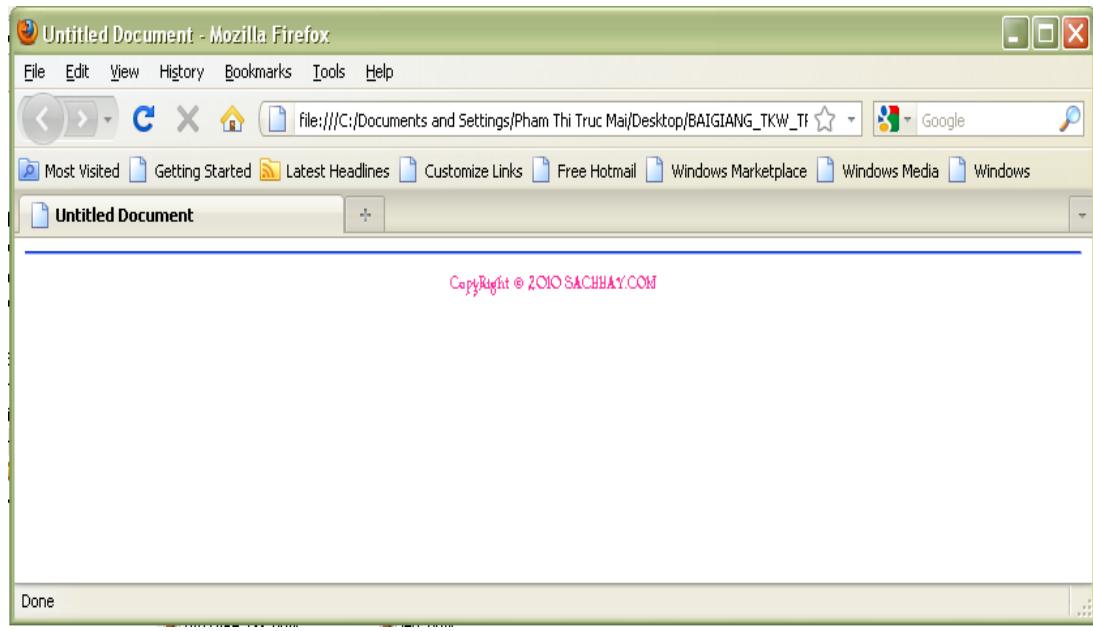
The main content area features a large blue header "CHỦ ĐỀ VĂN HỌC". Below it is a table listing two books:

Tên sách	Hình ảnh	Mô tả	Tác giả	Trang	Giá (VNĐ)
Mộng Đời Bất Tuyệt		Cuốn sách "Mộng Đời Bất Tuyệt (Tùy Bút)" giới thiệu nội dung: Còn Nhớ Tiếng Ve sầu?, Hoa Trên Cát	Nguyễn Tường Bách	500	56.000
Vượt Qua Thủ Thách		"Vượt Qua Thủ Thách" giới thiệu đến bạn đọc những câu chuyện như: Chiếc đai lưng màu xanh tía. Một chiếc cầu vòng vẫn không đủ.	Minh Hằng	700	70.000

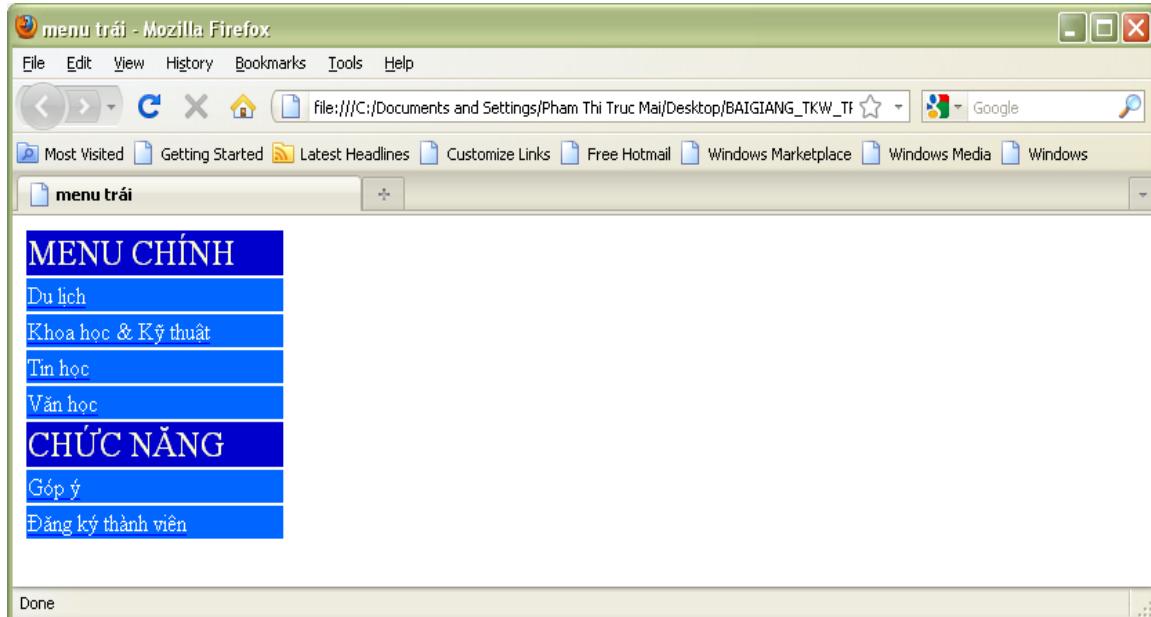
8. Hãy thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap15.html:



9. Hãy thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap16.html:



10. Hãy thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap17.html:



## BÀI 5

### **FORM VÀ CÁC CONTROLS**

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Tạo Form và các Controls

#### **1.5.1 Ý nghĩa của form trong trang Web và cách khởi tạo**

Web không chỉ dừng lại ở việc hiển thị thông tin mà nó còn có thể cho phép người dùng tương tác trên nó. Ví dụ: người dùng có thể nhập thông tin lên biểu mẫu và gửi về máy chủ xử lý. Trong khuôn khổ của việc thiết kế Web tĩnh, chúng tôi xin giới thiệu thẻ `<form>` ... `</form>` là một thẻ HTML cung cấp các biểu mẫu nhập liệu và các thành phần bên trong nó.

Cấu trúc chung:

```
<form name= "ten_form" method= "POST/GET" action= "URL">  
    <!–Các thành phần điều khiển trong form -->  
</form>
```

Trong đó:

*Name:* dùng để đặt tên cho form. Nếu form không được đặt tên, sẽ không thể can thiệp vào form bằng JavaScript để kiểm tra dữ liệu nhập có hợp lệ hay không. Tên form không được đặt có dấu tiếng Việt và không có dấu khoảng cách.

*Method:* xác định phương thức gửi dữ liệu về máy chủ. Nếu phương thức là *POST*, các thông tin nhập liệu của người dùng sẽ không hiển thị trên thanh địa chỉ của trình duyệt. Ngược lại, nếu phương thức gửi là *GET* thì các thông tin mà người dùng gửi đi sẽ được nhìn thấy trên thanh địa chỉ của trình duyệt. Mặc định phương thức truyền dữ liệu là *GET*. Do đó, vì vấn đề bảo mật, nếu trường hợp biểu mẫu nhập liệu có những thông tin cá nhân, nhưng thông tin cần được giữ bí mật như mật khẩu, mã PIN, ... thì phải dùng phương thức *POST*.

*Action:* trong thuộc tính này có thể là địa chỉ của trang sẽ nhận thông tin người dùng nhập để xử lý khi được gửi về máy chủ, hoặc cũng có thể để rỗng nếu bạn muốn thông tin khi gửi về máy chủ sẽ được xử lý tại chính trang biểu mẫu này.

Lưu ý: Nếu khai báo các thành phần điều khiển bên ngoài thẻ `<form> ... </form>` hoặc bỏ qua việc khai báo `<form> ... </form>` thì các thông tin mà người dùng nhập sẽ không thể gửi về Web Server. Điều này có nghĩa là các thông tin người dùng nhập sẽ không được xử lý.

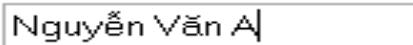
Chúng ta có thể vận dụng thẻ `<form>` để tạo các biểu mẫu như đăng nhập, đăng ký, viết bình luận, góp ý, ...

### 1.5.2 Các thành phần trong form

Tương tự như quy ước khi đặt tên form, tên của các thành phần trong form cũng không được đặt có dấu tiếng Việt và không có dấu khoảng cách. Sau đây, chúng tôi xin giới thiệu các thành phần cơ bản thường dùng trong form.

#### 1.5.2.1 Thẻ `<input>`

Đây là một thẻ đơn và tùy theo giá trị gọi trong thuộc tính `type` của thẻ này mà có thể tạo ra nhiều dạng khác nhau.

Text Field là một dạng khung nhập liệu chỉ cho phép người dùng nhập vào một dòng ngắn (`type="text"`).  Nếu muốn xác định chiều dài tối đa của chuỗi nhập bạn dùng thuộc tính `maxlength`. Nếu muốn tùy chỉnh kích thước của khung nhập liệu trên Web, bạn dùng thuộc tính `size`. Nếu muốn truyền một giá trị mặc định khi tải trang Web lên sẽ xuất hiện ngay giá trị này, bạn có thể sử dụng thuộc tính `value`. Nếu bạn muốn các giá trị trong Text Field ở dạng chỉ cho phép đọc, bạn gọi thuộc tính `readonly`. Bạn cũng có thể đặt một `id` nhận dạng cho Text Field. Tuy nhiên, để bót rườm rà và không mất nhiều thời gian, người ta thường quan trọng thuộc tính `name` hơn. Bạn chỉ cần đặt tên của các thành phần của form khác tên nhau thì không cần dùng đến `id` nữa.

Cú pháp:

```
<input type="text" name="ten_the" maxlength="n" size="m"
value="giá trị mặc định" readonly="readonly"
id="id_nhan_dang">
```

Password Field cũng là một dạng khung nhập liệu chỉ cho phép người dùng nhập vào một dòng ngắn nhưng đặc biệt hơn Text Field ở chỗ nó chuyển các ký tự mà người dùng nhập sang dạng dấu \*.  Password Field được dùng cho trường hợp

nhập các dữ liệu mật. Các thuộc tính trong nó tương tự như trong Text Field.

Cú pháp:

```
<input type="password" name="ten_the" maxlength="n" size="m"  
value="giá trị mặc định" readonly="readonly"  
id="id_nhan_dang">
```

Radio Button là một dạng cho phép chọn duy nhất 1 trong nhiều lựa chọn, dấu lựa chọn của Radio Button có dạng tròn  . Nếu muốn một lựa chọn nào đó được chọn mặc định, bạn dùng thuộc tính *checked*. Bên trong thuộc tính *value* là giá trị tương ứng sẽ được gửi đi khi người dùng chọn một lựa chọn. Chú ý rằng nếu các lựa chọn mà bạn trình bày thuộc cùng một nhóm lựa chọn, bạn phải đặt các lựa chọn này cùng tên để có thể chọn duy nhất 1 lựa chọn.

Cú pháp:

```
<input type="radio" name="ten_the" checked="checked"  
value="giá trị" id="id_nhan_dang">
```

Checkbox là dạng cho phép chọn nhiều hơn 1 lựa chọn, dấu lựa chọn của Checkbox có dạng vuông  . Các thuộc tính bên trong Checkbox cũng tương tự như các thuộc tính của Radio Button. Chú ý rằng nếu các lựa chọn mà bạn trình bày thuộc cùng một nhóm lựa chọn, bạn cũng phải đặt các lựa chọn này cùng tên.

Cú pháp:

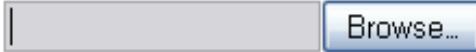
```
<input type="checkbox" name="ten_the" checked="checked"  
value="giá trị" id="id_nhan_dang">
```

Hidden Text Field là một dạng cho phép văn bản ẩn. Điều này có nghĩa là người thiết kế Web muốn truyền ngầm một nội dung nào đó mà người dùng không cần biết có sự tồn tại của nội dung đó. Các thuộc tính bên trong thẻ này hoàn toàn giống các thuộc tính của Text Field. Tuy nhiên, do đây là một dạng thẻ ẩn nội dung nên người ta không quan tâm nhiều đến các thuộc tính định dạng mà chỉ quan tâm đến các thuộc tính *type* và *name*.

Cú pháp:

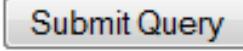
```
<input type="hidden" name="ten_the">
```

File Field là một khung cho phép chọn một hình ảnh từ máy người dùng để tải lên

Web  . Để sử dụng được loại thẻ này, bạn cần thêm thuộc tính *enctype=“multipart/form-data”* vào phần khai báo thẻ *<form>* và để truyền được một tập tin từ máy người dùng lên máy chủ, bạn còn cần dùng thêm dạng Hidden Text Field để truyền ngầm dung lượng của tập tin.

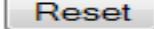
Cú pháp:

```
<input type="file" name="ten_the">
```

Submit Button là một dạng nút nhấn, thường có trên form với vai trò đánh dấu kết thúc việc nhập liệu và tiến hành gửi dữ liệu về máy chủ. 

Cú pháp:

```
<input type="submit" name="ten_the" value= "Giá trị hiện trên nút">
```

Reset Button là nút có chức năng hủy bỏ các dữ liệu mà người dùng vừa nhập vào form. 

Cú pháp:

```
<input type="reset" name="ten_the" value= "Giá trị hiện trên nút">
```

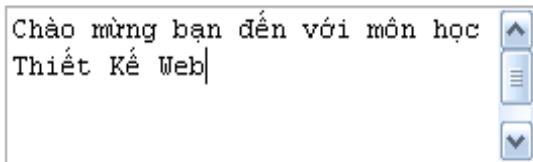
Button là một dạng nút thường  . Nút này sẽ thực hiện chức năng khác nhau tùy thuộc vào ý định của người thiết kế. Thông thường, nút này được dùng để gọi thực thi một lệnh / hàm trong JavaScript thông qua các thuộc tính sự kiện như *onClick* / *onChange* / *onBlur* / ... Chúng tôi sẽ nói rõ hơn về các thuộc tính sự kiện này ở chương JavaScript.

Cú pháp:

```
<input type="button" name="ten_the" value= "Giá trị hiện trên  
nút">
```

#### 1.5.2.2 *Thẻ <textarea> ... </textarea>*

Khắc phục hạn chế của Text Field, *<textarea>* cho phép nhập nội dung trên nhiều dòng. Ví dụ:



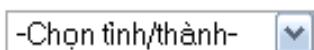
Bạn phải xác định độ lớn của khung Textarea bằng hai thuộc tính số dòng (*rows*) và số cột (*cols*). Nếu muốn hiển thị lên khung Textarea chuỗi văn bản mặc định, bạn nhập chuỗi giữa cặp thẻ *<textarea> ... </textarea>*. Nếu muốn chuỗi này chỉ được phép đọc, người dùng không thể chỉnh sửa thì bạn thêm thuộc tính *readonly* như trong Text Field.

Cú pháp:

```
<textarea name="ten_the" rows="n" cols="m"> </textarea>
```

#### 1.5.2.3 *Thẻ <select> ... </select>*

Đây là một dạng lựa chọn dưới hình thức danh sách lựa chọn. Nếu bạn muốn người dùng chỉ chọn được 1 lựa chọn tại một thời điểm thì đó chính là Combobox. Ví dụ:



Nếu bạn muốn người dùng chọn được nhiều lựa chọn trong danh sách tại cùng một thời điểm thì đó chính là Listbox. Trường hợp là Listbox, bạn thêm thuộc tính *multiple* vào trong thẻ *<select>*. Ví dụ:



Mỗi một phần tử trong danh sách được định nghĩa trong cặp thẻ *<option> ...*

`</option>`. Để trình duyệt cũng như Web Server hiểu được các phần tử này khác nhau như thế nào, bạn cần chỉ định giá trị của mỗi phần tử trong thuộc tính `value` của thẻ `<option>`. Nếu bạn muốn chỉ định phần tử được chọn mặc định, dùng thuộc tính `selected = "selected"`. Trong Combobox, phần tử được chọn mặc định sẽ là phần tử duy nhất được nhìn thấy khi form được tải lên do bản thân Combobox là dạng ẩn các phần tử khi chưa được tương tác. Trong Listbox, các phần tử được chọn mặc định sẽ được tô màu nền.

Cú pháp Combobox:

```
<select name="ten_combobox">
    <option value= "giá trị 1">Tiêu đề phần tử 1</option>
    <option value= "giá trị 2">Tiêu đề phần tử 2</option>
    <option value= "giá trị n">Tiêu đề phần tử n</option>
</select>
```

Cú pháp Listbox:

```
<select name="ten_combobox" multiple>
    <option value= "giá trị 1">Tiêu đề phần tử 1</option>
    <option value= "giá trị 2">Tiêu đề phần tử 2</option>
    <option value= "giá trị n">Tiêu đề phần tử n</option>
</select>
```

#### 1.5.2.4 Thẻ `<fieldset> ... </fieldset>`

Đây là thẻ dùng để nhóm các thành phần nhập liệu. Cặp thẻ này chỉ có tác dụng trang trí cho form.



Khi xác định các thành phần nhập liệu thuộc cùng một nhóm, bạn gọi các thành phần đó bên trong cặp thẻ `<fieldset> ... </fieldset>`

Cú pháp:

```
<fieldset>
    <legend> Tiêu đề nhóm </legend>
    <input ...>
    ...
</fieldset>
```

### Ví dụ 1-5-1: Tổng hợp các thành phần trong form

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Ví dụ 1-5-1</title>
</head>
<body>
<h1 align="center"> ĐĂNG KÝ THÀNH VIÊN </h1>
<HR align="center" width="250" />
<FORM action="" name="dangnhap" method="post">
Họ tên của bạn: <input type="text" name="txtHTen" size="20"
/>
<br>
Mã số sinh viên: <input type="text" name="txtMSSV" size="19"
maxlength="9" />
<br>
Mật khẩu: <input type="password" name="pswMKhau" size="15"
maxlength="10" />
<br>
Quê quán: <select name="slQQuan">
            <option value="CT">Cần Thơ</option>
            <option value="DT">Đồng Tháp</option>
            <option value="TV" selected="selected">Trà
Vinh</option>
            <option value="VL">Vĩnh Long</option>
        </select>
<br>
Những môn đã tích lũy trên 8.5:
<select name="slMon" multiple>
    <option value="Thiết kế Web" selected="selected">Thiết
kế Web</option>
    <option value="Cơ sở dữ liệu">Cơ sở dữ liệu</option>
    <option value="Phân tích và thiết kế HTTT">PT&TK
HTTT</option>
    <option value="Đồ họa ứng dụng" selected="selected">Đồ
họa ứng dụng</option>
</select>
<fieldset>
    <legend>Giới tính</legend>
    <input type="radio" name="rdGTinh" value="0" />Nam
    &nbsp;&nbsp;
    <input type="radio" name="rdGTinh" value="1" /> Nữ
</fieldset>
<p>
Sở thích: <br>
<pre>      <input type = "checkbox" name="chbxSThich"
value="phim" />Xem phim, ca nhạc
</pre>
```

```

<input type="checkbox" name="chbxSThich" value="docbao"
/> Đọc báo
<input type="checkbox" name="chbxSThich" value="luotweb"
/> Luôt Web
</pre>
<textarea name="txtaQDinh" cols="80" rows="5"
readonly="readonly">
- Diễn đàn sẽ không chấp nhận việc lợi dụng diễn đàn làm nơi
trao đổi các thông tin phản động hay các văn hóa phẩm đồi
trị.
- Nếu Ban quản trị phát hiện tài khoản của bạn có các hoạt
động khả nghi, chúng tôi có quyền xóa tài khoản của bạn mà
không cần báo trước.
</textarea><br>
<input type="submit" name="sbmDNhap" value="Đăng ký" />
<input type="reset" name="rsLLai" value="Làm lại" />
</FORM>
</body>
</html>

```

**ĐĂNG KÝ THÀNH VIÊN**

Họ tên của bạn:

Mã số sinh viên:

Mật khẩu:

Quê quán:  Trà Vinh

Những môn đã tích lũy trên 8.5:  **Đồ họa ứng dụng**

Giới tính:

Nam  Nữ

Sở thích:

Xem phim, ca nhạc  
 Đọc báo  
 Luôt Web

- Diễn đàn sẽ không chấp nhận việc lợi dụng diễn đàn làm nơi trao đổi các thông tin phản động hay các văn hóa phẩm đồi trì.  
- Nếu Ban quản trị phát hiện tài khoản của bạn có các hoạt động khả nghi, chúng tôi có quyền xóa tài khoản của bạn mà không cần báo trước.

**Đăng ký** **Làm lại**

**Hình 1-5-1: Trang Vidu1-5-1.html**

Chúng ta đã được giới thiệu rất nhiều thành phần điều khiển (controls) của form, như vậy khi nào chúng ta chọn thành phần này mà không chọn thành phần kia. Chúng tôi xin tập trung phân tích từng thành phần của form trong ví dụ 1-5-1 này.

Đầu tiên, đây là một biểu mẫu yêu cầu người duyệt Web cung cấp các thông tin cần thiết trong đó có mật khẩu để trở thành thành viên của Website. Do đó, thuộc tính *method*= “post” là điều bắt buộc. Chúng tôi đặt cho form một cái tên *name*= “*dangnhap*” để phân biệt với các form khác (nếu có) trong một Website. Như đã trình bày, bên trong *name* chúng ta không dùng dấu khoảng cách và từ tiếng Việt có dấu. Bên cạnh đó, chúng tôi đang thực hiện tạo giao diện form nên chưa có tập tin xử lý cho form này. Vì vậy, thuộc tính *action* được để rỗng.

Thứ hai, phần nhập họ tên sinh viên, chúng tôi chọn *type*= “text” cho thẻ *<input>* vì tên của một người thì không quá dài, ta không cần phải chọn dạng nhập nhiều dòng là *<textarea>*. Để phân biệt Text Field này với các Text Field khác trong form, chúng tôi đặt cho nó một cái tên *name*= “*txtHTen*”. Để có thể hiển thị đủ tên của một người trên khung nhập liệu, chúng tôi quy định thuộc tính *size*= “20” cho thẻ này.

Thứ ba, tương tự cho trường nhập mã số sinh viên. Tuy nhiên như đã biết, mã số sinh viên của một sinh viên thuộc trường Đại học Trà Vinh có 9 ký số. Để hạn chế lỗi phát sinh trong quá trình nhập liệu, chúng ta cần quy định số ký tự tối đa cho trường nhập liệu này là 9, tức là *maxlength*= “9”.

Thứ tư, trường nhập mật khẩu là một kiểu nhập liệu đặc biệt, những gì người duyệt Web nhập vào trường này đều được chuyển thành các ký tự \* nhằm ngăn chặn người khác nhìn thấy. Do đó, *type* ở đây phải chọn là *password*. Tùy theo ý định của người thiết kế form mà có thể quy định chiều dài tối đa của mật khẩu. Ở đây chúng tôi quy định chiều dài tối đa của chuỗi mật khẩu là 10 ký tự.

Thứ năm, giải thích mục lựa chọn quê quán. Hiện nay nước Việt Nam chúng ta có đến 63 tỉnh / thành phố. Việc hiển thị tất cả chúng ra sẽ chiếm rất nhiều không gian. Mặt khác, mỗi người chúng ta chỉ có duy nhất một nơi được gọi là quê quán. Do đó, việc chọn thẻ *<select>* ở đây là hoàn toàn hợp lý. Mỗi tỉnh thành đều được đặt trong một *<option>* riêng. Tương ứng với mỗi *<option>* lựa chọn sẽ được truyền đi một giá trị bên trong thuộc tính *value*. Ví dụ, khi người duyệt Web chọn quê quán là “Cần Thơ” thì giá trị sẽ được gửi đi xử lý sẽ là “CT”. Ở *<option>* lựa chọn tỉnh Trà Vinh, chúng tôi thêm thuộc tính

*selected=“selected”* để chọn Trà Vinh làm tỉnh hiển thị mặc định bên trong Combobox.

Thứ sáu, về mục chọn những môn đã tích lũy trên 8.5, ở đây chúng tôi chỉ dùng 4 môn học làm dữ liệu mẫu. Do một sinh viên có thể đạt trên 8.5 điểm ở nhiều môn khác nhau nên bắt buộc chúng ta phải dùng hộp lựa chọn nhiều. Tuy nhiên, nếu muốn, bạn có thể thay thế thẻ *<select>* dạng *multiple* thành thẻ *<input>* có *type=“checkbox”*. Đây là hai cách hiển thị lựa chọn nhiều, bạn có thể linh hoạt để bố trí sao cho Website của mình phù hợp với mục đích thiết kế. Trong 4 môn hiển thị mẫu thì có 2 môn được chọn mặc định là Thiết kế Web và Đồ họa ứng dụng. Như vậy, tại các *<option>* của 2 lựa chọn này, chúng tôi gọi thuộc tính *selected=“selected”*. Trường hợp bạn cho hiển thị dạng checkbox thì cần lưu ý đặt các thẻ *<input type=“checkbox”>* này cùng *name*, khác nhau ở *value* và gọi *checked=“checked”* tại các thẻ *<input>* giữ giá trị của 2 môn Thiết kế Web và Đồ họa ứng dụng.

Thứ bảy, phần *<fieldset>* được dùng để gồm nhóm lựa chọn giới tính khi trình bày trên giao diện.

Thứ tám, về phần chọn giới tính. Chúng ta chỉ cho phép chọn mỗi lần 1 giới tính Nam hoặc Nữ, do đó đây là loại có một lựa chọn. Chúng ta có thể dùng dạng *<select>* hoặc dạng *<input>* có *type=“radio”*. Ở đây chúng tôi chọn dạng *radio* để việc trình bày được đa dạng hơn. Tương ứng khi người duyệt Web chọn Nam, giá trị truyền bên trong sẽ là 0. Ngược lại nếu người duyệt Web chọn giới tính là Nữ thì giá trị sẽ truyền đi là 1. Để hai lựa chọn Nam và Nữ chỉ được chọn 1 thì bắt buộc thuộc tính *name* phải đặt trùng nhau. Ở đây chúng tôi đặt *name=“rdGTinh”*.

Thứ chín, tương tự như phần chọn các môn đã tích lũy trên 8.5, chúng ta có thể sử dụng *<select>* dạng *multiple* hoặc dùng thẻ *<input type=“checkbox”* cho phần chọn sở thích vì một người có thể có nhiều sở thích. Ở đây, chúng tôi chọn dạng *checkbox*. Các mục chọn Xem phim – ca nhạc, Đọc báo hay Lướt Web đều thuộc nhóm sở thích, do đó chúng ta cần đặt *name* của chúng trùng nhau, trong ví dụ này là *name=“chbxSThich”*.

Thứ mười là phần nội quy diễn đàn, chúng ta có thể viết trực tiếp trên trang Web. Tuy nhiên, nếu muốn trình bày nội quy này trên form thì chúng ta có thể chọn dạng thẻ *<textarea>* vì nó cho phép hiển thị nội dung trên nhiều dòng. Tuy nhiên, nội quy là những quy định mà chúng ta bắt buộc người dùng khi là thành viên phải tuân theo, và đó là điều mà người dùng chỉ được phép đọc mà không được phép chỉnh sửa. Vì vậy thuộc tính

`readonly=“readonly”` là rất cần thiết cho trường hợp này.

Thứ mười một, sau khi người duyệt Web đã nhập các thông tin cần thiết thì cần có một nút nhấn đóng vai trò quyết định báo kết thúc quá trình nhập liệu và gửi các thông tin của form nhận được từ phía người dùng lên máy chủ. Đó chính là nút Submit. Trong trường hợp này, nút Submit có nhãn / giá trị hiển thị trên nút là Đăng ký, chúng ta dùng `value=“Đăng ký”` cho thẻ `<input type=“submit”>`.

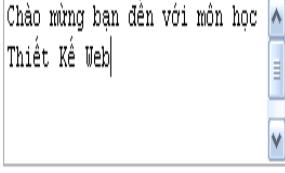
Cuối cùng, trong trường hợp người duyệt Web muốn xóa bỏ các thông tin vừa nhập để nhập lại thông tin khác bằng cách nhấn vào nút Làm lại. Chúng ta gọi thêm một thẻ `<input type=“reset”>` và chỉ định `value=“Làm lại”` cho nó.

Như vậy, chúng tôi đã giải thích xong ý nghĩa cũng như giải thích lý do chọn từng thành phần cho form trong ví dụ 1-5-1. Trên thực tế có rất nhiều cách trình bày form. Một trong những cách phổ biến nhất đó là cho một bảng nằm trong form để chia form nhập liệu thành 2 cột: một cột hiện yêu cầu nhập liệu, một cột hiện các thành phần của form. Chúng ta sẽ có nhiều bài tập vận dụng cách trình bày này trong phần bài tập cuối có bên dưới.

#### 1.5.2.5 Bảng tổng hợp các thành phần trong form

TÊN THẺ	THUỘC TÍNH CHÍNH	THUỘC TÍNH PHỤ	NHẬN DIỆN	TÁC DỤNG
<code>&lt;input&gt;</code>	<code>type=“text”</code>	<code>Name= “ten_the”</code> <code>Value= “giá trị”</code> <code>Maxlength= “n”</code> <code>Size= “m”</code> <code> Readonly=“readonly”</code>	<input type="text" value="Nguyễn Văn A "/>	Nhập liệu 1 dòng ngắn
	<code>type=“password”</code>	<code>Name= “ten_the”</code> <code>Value= “giá trị”</code> <code>Maxlength= “n”</code> <code>Size= “m”</code>	<input type="password" value="•••• "/>	Dữ liệu nhập sẽ ẩn bên trong dấu *

TÊN THẺ	THUỘC TÍNH CHÍNH	THUỘC TÍNH PHỤ	NHẬN DIỆN	TÁC DỤNG
	<code>type="radio"</code>	<code>name="ten_the"</code> <code>value="giá trị"</code> <code>checked="checked"</code>	<input checked="" type="radio"/>	Chọn 1 trong nhiều
	<code>type="checkbox"</code>	<code>name="ten_the"</code> <code>value="giá trị"</code> <code>checked="checked"</code>	<input type="checkbox"/>	Chọn nhiều trong nhiều
	<code>type="file"</code>		<input type="file"/> Browse...	Chọn một tập tin từ máy tính
	<code>type="hidden"</code>	<code>name="ten_the"</code> <code>value="giá trị"</code>		Truyền những giá trị ẩn
	<code>type="submit"</code>	<code>name="ten_the"</code> <code>value="giá trị"</code>	<b>Submit Query</b>	Nút gửi thông tin về máy chủ
	<code>type="reset"</code>	<code>name="ten_the"</code> <code>value="giá trị"</code>	<b>Reset</b>	Nút hủy bỏ dữ liệu
	<code>type="button"</code>	<code>name="ten_the"</code> <code>value="giá trị"</code> <code>onClick / onChange</code> / <code>onBlur="lệnh Javascript"</code>	<b>Button</b>	Nút nhấn thực thi theo ý định lập trình của người thiết kế

TÊN THẺ	THUỘC TÍNH CHÍNH	THUỘC TÍNH PHỤ	NHẬN DIỆN	TÁC DỤNG
<textarea> --- </textarea> >	rows="n" cols="m"	name="ten_the" readonly="readonly"		Nhập nhiều dòng
<select> ... </select>	Multiple	name="ten_the"		Chọn 1 trong nhiều
<option> ... </option>		value="giá trị"		Từng lựa chọn trong Combobox hoặc Listbox
<fieldset> ... </fieldset>				Gôm nhóm trên form
<legend> ... </legend>		align="left / right / top / bottom"		Tiêu đề nhóm

❖ Câu hỏi (bài tập) cung cấp:

- Hãy thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap18.html:

The screenshot shows a Mozilla Firefox browser window with the title bar "DĂNG KÝ THÀNH VIÊN - Mozilla Firefox". The address bar displays the URL "file:///C:/Documents and Settings/Pham Thi Truc Mai/Desktop/BAIGIANG". The main content area contains a registration form with the following fields:

- Tên đăng nhập: (\*)
- Mật khẩu: (\*)
- Nhập lại mật khẩu: (\*)
- Ngày sinh: -ngày- [dropdown] -tháng- [dropdown] năm [dropdown]
- Giới tính:  Nam  Nữ
- Email: [text input]
- Tỉnh/Thành phố: -Chọn tỉnh/thành- [dropdown]
- Câu châm ngôn của bạn: [text area]

At the bottom of the form are two buttons: "Đăng ký" (Register) and "Làm lại" (Reset). A "Done" button is also visible at the very bottom.

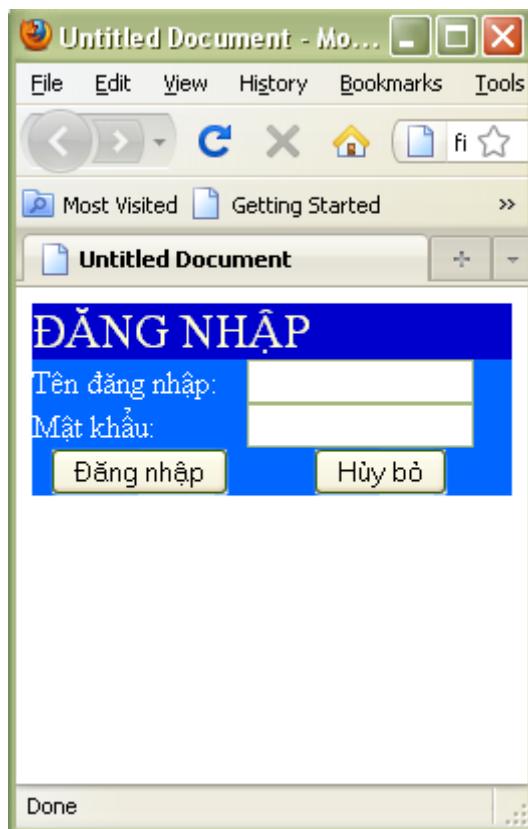
2. Hãy thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap19.html:

The screenshot shows a Mozilla Firefox browser window with the title bar "TRANG GOP Y - Mozilla Firefox". The address bar displays "file:///C:/Documents and Settings/Pham Thi Truc Mai/Desktop/BAIGIANG". The toolbar includes standard buttons for back, forward, search, and refresh, along with links to "Google" and "Customize Links". Below the toolbar is a menu bar with "File", "Edit", "View", "History", "Bookmarks", "Tools", and "Help". A bookmarks bar contains links to "Most Visited", "Getting Started", "Latest Headlines", "Customize Links", "Free Hotmail", "Windows Marketplace", and "Windows Media". The main content area features a large blue header "HỘP THƯ GÓP Ý". Below it is a form with the following fields:

- Họ tên: (Text input field)
- Quê quán: (Dropdown menu with placeholder "-Chọn tỉnh/thành-")
- Góp ý về lĩnh vực: (Checkboxes for: Bộ cục, Giao diện Web, Thể loại sách)
- Ngôn ngữ: (Radio buttons for: Anh, Việt, with Vietnamese selected)
- Nội dung: (Large text area)
- Gửi (Green button)
- Hủy bỏ (Green button)

The status bar at the bottom left says "Done".

3. Hãy thiết kế trang Web có nội dung như sau. Lưu bài tập vào D:/hotenSV/ với tên baitap20.html:



4. Hãy thiết kế trang Web có nội dung như sau bằng frameset. Lưu bài tập vào D:/hotenSV/ với tên Tonghop1.html:

SACHHAY.COM - Mozilla Firefox

File Edit View History Bookmarks Tools Help

SACHHAY.COM

file:///D:/DHTV/BAI GIANG/BAIGIANG\_TKW\_TRUCMAI/TH\_TKWEB/ontap\_trangchu.html

Google

Disable Cookies CSS Forms Images Information Miscellaneous Outline Resizer Tools View Source Options

Chào mừng bạn đến với website giới thiệu sách! Rất vui được phục vụ!!!

MENU CHÍNH

Du lịch

Khoa học & Kỹ thuật

Tin học

Văn học

CHỨC NĂNG

Góp ý

Đăng ký thành viên

Tự Học Javascript

Javascript là một ngôn ngữ lập trình chính của Web và cũng là ngôn ngữ dễ nắm bắt nhất. Có một số lý do khiến bạn tiếp cận ngôn ngữ này một cách dễ dàng: Javascript Vòng

JavaCript

HTML và bạn có thể bắt đầu Nước Hàn bằng cách đưa thủ tục nhỏ Quốc gọn...

Các Phương Pháp Thi Công Xây Dựng

"Vuot Qua Thủ Thách" giới thiệu đến bạn đọc những câu chuyện như: Chiếc đại lung

ĐĂNG NHẬP

Tên đăng nhập:

Mật khẩu:

Đăng nhập Hủy bỏ

Copyright © 2010 SACHHAY.COM

5. Hãy thiết kế trang Web có nội dung như sau bằng frameset. Lưu bài tập vào D:/hotenSV/ với tên Tonghop2.html:

Tên sách	Hình ảnh	Mô tả	Tác giả	Trang	Giá (VND)
Vòng Quanh Nước Hàn Quốc		Cuốn sách Một Vòng Quanh Nước Hàn Quốc (Du Lịch & Du học) giới thiệu lịch sử, địa lý, kinh tế, văn hóa - xã hội, du lịch, giáo dục, du học của nước Hàn Quốc	Trần Bão Quốc	650	72.000
Du Lịch Châu Âu		Cuốn sách "Du Lịch Châu Âu" là một trong những tập nằm trong Bộ sách "Du Lịch Thế Giới". Sách giới thiệu đến bạn đọc một số nước du lịch: Phùng Iceland, Na Uy...	Phùng Văn Hòa	412	38.000
Du Lịch Châu Á		Cuốn "Du Lịch Châu Á" là một trong những tập nằm trong Bộ sách "Du Lịch Thế Giới". Nội dung cuốn sách giới thiệu một số nước là điểm du lịch trong khu vực Châu Á: Hàn Quốc, Trung Quốc	Phùng Văn Hòa	620	65.000

6. Hãy thiết kế trang Web có nội dung như sau bằng frameset. Lưu bài tập vào D:/hotenSV/ với tên Tonghop3.html:

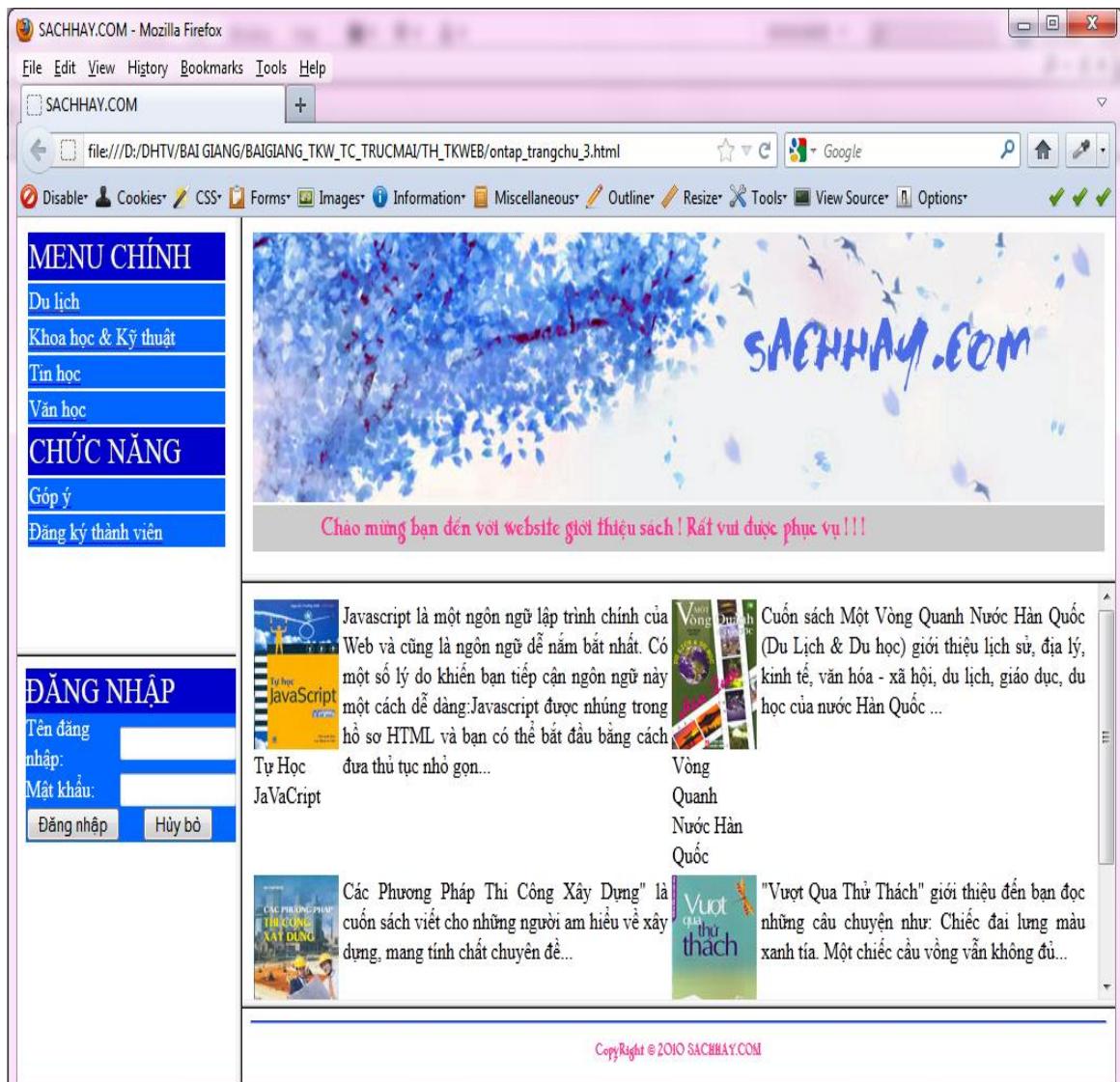
Javascript là một ngôn ngữ lập trình chính của Web và cũng là ngôn ngữ dễ nắm bắt nhất. Có một số lý do khiến bạn tiếp cận ngôn ngữ này một cách dễ dàng: Javascript được nhúng Vòng Tròn trong hồ sơ HTML và bạn có thể bắt đầu bằng cách đưa thử Nước Hàn Quốc...

Tự Học JavaScript

Các Phương Pháp Thi Công Xây Dựng" là cuốn sách viết cho những người am hiểu về xây dựng, mang tính chất chuyên đề...

Vượt Qua Thủ Thách

7. Hãy thiết kế trang Web có nội dung như sau bằng frameset. Lưu bài tập vào D:/hotenSV/ với tên Tonghop4.html:



8. Hãy thiết kế lại các trang giao diện trên bằng table.

## CHƯƠNG 2

### BẢNG MẪU NẠP CHỒNG – CSS

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Vận dụng CSS thiết kế giao diện

#### 2.1 Khái niệm

CSS - Cascading Style Sheet – Bảng mẫu nạp chồng quy định cách trình bày cho các tài liệu HTML, XHTML, XML, ...

Ở đây xin nói cụ thể về ngôn ngữ mà chúng ta vừa tìm hiểu qua, HTML. Như bạn đã biết, HTML cũng đã hỗ trợ khá nhiều thẻ và các thuộc tính bên trong để định dạng cho các đối tượng như văn bản, hình ảnh, bảng biểu, ... Tuy nhiên, chắc hẳn bạn cũng nhận thấy một điều rằng đôi khi để có được một định dạng cơ bản trên Web, chúng ta cũng đã phải kết hợp rất nhiều thẻ HTML để đạt được ý định hiển thị. Đó là chưa kể đến quá trình dịch trang Web sẽ bị chậm đi khi sử dụng quá nhiều mã. Đó cũng chưa kể đến việc mã nguồn phức tạp, rắc rối nếu bạn không có một kỹ thuật lập trình tốt thì rất khó để chỉnh sửa, cập nhật lại trang khi có thay đổi. Đó cũng chưa tính đến chuyện tốn hao nhiều thời gian để thiết kế, nếu bạn có một Website với nội dung lớn, có khá nhiều trang, rất có thể những mã thẻ mà bạn gọi sẽ khác nhau ở các trang. Vì vậy sẽ dẫn đến sự thiếu đồng nhất về định dạng cho toàn bộ giao diện của một Website.

Trong khi đó CSS sẽ hỗ trợ bạn khắc phục các nhược điểm vừa nêu. Do CSS là một dạng mẫu được quy định sẵn và được nạp vào bộ nhớ của trình duyệt nên ở những lần truy cập lại sẽ giảm được thời gian nạp và dịch lại định dạng. Từ một bảng mẫu mà bạn thiết kế có thể áp dụng cho nhiều trang thậm chí cho các Website khác nhau. Đây là một ưu thế rõ rệt của CSS vì bạn không cần mất thời gian để thực hiện lại các thao tác định dạng giống nhau. Bên cạnh đó, hiện nay đa số các trình duyệt hiện đại đều có hỗ trợ hiển thị CSS. Điều duy nhất khiến người thiết kế Web lo lắng đó là để vận dụng tốt CSS vào thiết kế Web thì họ cần phải tốn khá nhiều thời gian trong việc nhớ, học hỏi và tích lũy các kiến thức về CSS.

CSS được vận dụng linh hoạt trong trang HTML với 2 kiểu viết áp dụng cho 3 loại CSS:

Kiểu viết 1:

```
Selector {  
    property1: value1;  
    ...  
    propertyN: valueN;  
}
```

Trong đó:

Selector – bộ chọn, có thể là tên thẻ HTML, có thể là id của thẻ HTML hoặc một lớp được định dạng chung một số thẻ HTML, cũng có thể là một nhóm thẻ HTML lồng nhau.

Property là các thuộc tính của CSS, với value là giá trị của thuộc tính đó. Các thuộc tính trong CSS cách nhau bằng dấu chấm phẩy (;

Kiểu viết 2:

```
<tagname ... style="property1: value1;...; propertyN: valueN;">
```

Với kiểu 2, CSS được gọi trực tiếp trong một thẻ HTML nào đó thông qua thuộc tính *style*. Thay vì phải sử dụng quá nhiều thẻ HTML để định dạng cho một nội dung thì bạn chỉ cần gọi CSS là đủ, trong CSS có hỗ trợ hầu hết các định dạng cho một thẻ HTML bất kỳ.

Để đánh dấu ghi chú trong CSS, ta dùng: /\* nội dung ghi chú \*/

## 2.2 Phân loại

CSS được phân thành 3 loại: Bảng kiểu trực tiếp (Inline style sheet), Bảng kiểu được nhúng vào trong tài liệu HTML (Internal style sheet), Bảng kiểu bên ngoài (External style sheet). Những loại này có thể được sử dụng đồng thời, với thứ tự ưu tiên giảm dần như sau: 1. Inline style sheet, 2. Internal style sheet, 3. External style sheet, 4. Browser default. Điều này có nghĩa là nếu trong cùng một thẻ HTML cùng được quy định một thuộc tính nào đó bằng Inline style sheet và Internal style sheet thì thuộc tính sẽ được lấy để hiển thị chính là thuộc tính được quy định trong Inline style sheet. Còn trường hợp cùng một thẻ HTML được quy định nhiều thuộc tính bằng các loại CSS khác nhau thì để hiển thị định dạng của thẻ HTML đó, trình duyệt sẽ tổng hợp tất cả các định dạng đó lại để hiển thị, trường hợp các thuộc tính chưa được quy định bởi CSS nào thì trình duyệt sẽ lấy quy định mặc định của mình để hiển thị.

### 2.2.1 *Inline style sheet*

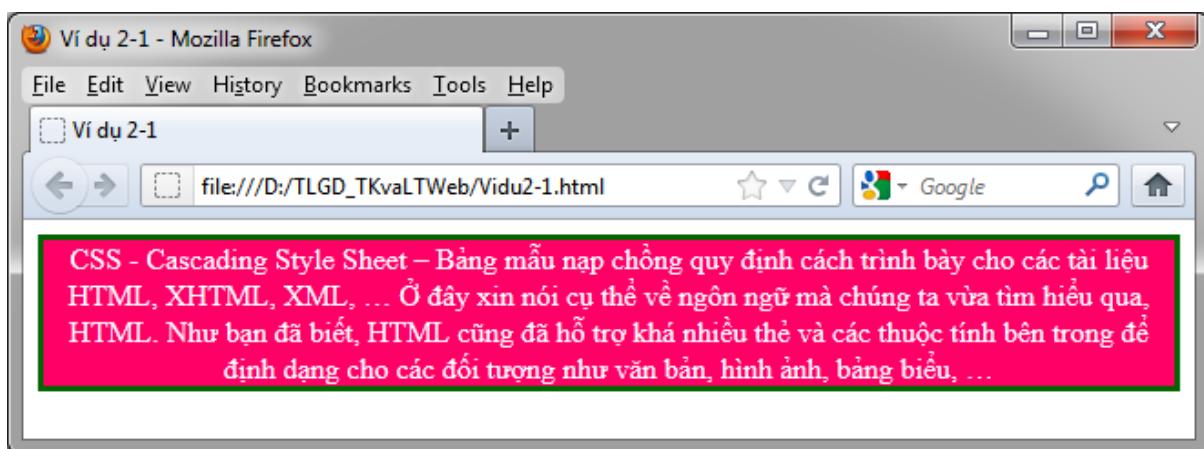
Đây là kiểu được gán cho một dòng hoặc một đoạn văn bản, CSS được viết trực tiếp bên trong thẻ HTML thông qua thuộc tính *style*.

Cú pháp:

```
<TagName Style=" property1:value 1; property2:value 2; ...;">  
    Nội dung văn bản muốn nhận định dạng từ thẻ  
</TagName>
```

Ví dụ 2-1: Bảng kiểu trực tiếp

```
<html>  
<head>  
<meta http-equiv="Content-Type" content="text/html;  
charset=utf-8" />  
<title>Ví dụ 2-1</title>  
</head>  
<body>  
<p align="center" style="background-color:#F06; color:#FFF;  
border:1px #F90 solid;">CSS - Cascading Style Sheet - Bảng  
mẫu nạp chồng quy định cách trình bày cho các tài liệu HTML,  
XHTML, XML, ... Ở đây xin nói cụ thể về ngôn ngữ mà chúng ta  
vừa tìm hiểu qua, HTML. Như bạn đã biết, HTML cũng đã hỗ trợ  
khá nhiều thẻ và các thuộc tính bên trong để định dạng cho  
các đối tượng như văn bản, hình ảnh, bảng biểu, ... </p>  
</body>  
</html>
```



Hình 2-1: Trang Vиду2-1.html

Trong ví dụ 2-1, đoạn văn bản được canh giữa do thuộc tính của thẻ *<p>*. Tuy nhiên trong *<p>* không có các thuộc tính màu chữ, màu nền, đường viền, màu của đường viền. Nếu chỉ dùng các thẻ HTML lồng nhau thì mã nguồn của đoạn này rất rườm rà. Còn trường hợp

này ta quy định thêm các thuộc tính mẫu có sẵn của CSS nên mã nguồn đơn giản hơn nhiều và dễ quản lý. Trong trường hợp ta cần định nghĩa cho các thuộc tính có liên quan nhau, như trong ví dụ này bạn thấy chúng tôi dùng `border:1px #F90 solid;` để định dạng cho độ dày của đường viền (`border-width`), mà của đường viền (`border-color`) và kiểu hiển thị đường viền (`border-style`). Thay vì phải viết: `border-width:3px; border-color:#060; border-style:solid;` CSS cho phép chúng ta gôm gọn định dạng thành: `border:1px #F90 solid;`.

**Ưu điểm của CSS loại 1:** Điều đầu tiên có thể thấy ngay đó là CSS loại 1 có hiệu lực ưu tiên cao nhất nên các thuộc tính gọi bằng CSS loại 1 luôn được ưu tiên hiển thị; điều thứ hai là chúng ta có thể dễ dàng quản lý theo từng thẻ của tài liệu HTML.

**Về nhược điểm:** Rõ ràng CSS loại 1 sẽ gây khó khăn cho các nhà phát triển Web trong việc cập nhật và mất nhiều thời gian khi phải định dạng đi định dạng lại cho cùng một thẻ HTML với cùng những thuộc tính như nhau.

### 2.2.2 Internal style sheet

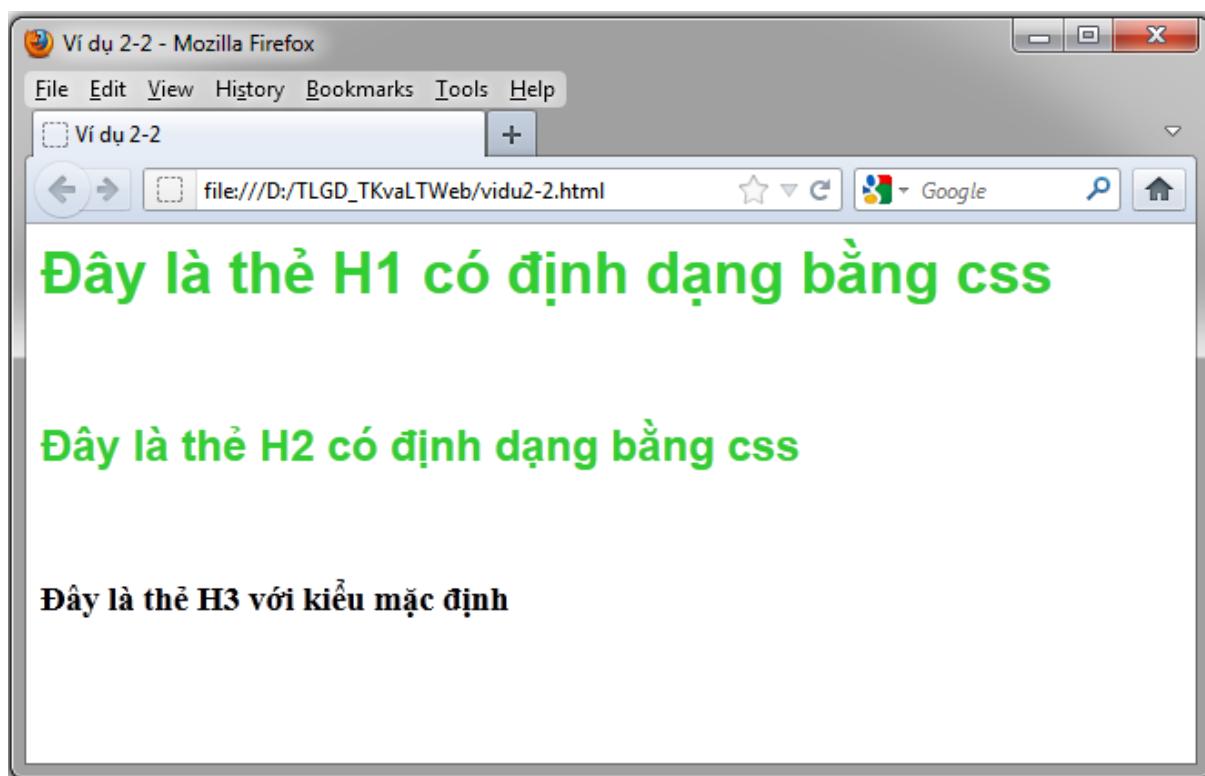
Là bảng mẫu thích hợp cho trang riêng lẻ với nhiều văn bản. Để tạo CSS loại này ta định nghĩa sẵn bảng mẫu chung trên phần đầu trang trong cặp thẻ `<head> ... </head>`. Ta áp dụng kiểu viết 1 cho CSS loại 2. Sau đó chỉ việc gọi các thẻ HTML trong `<body>` bình thường như đã biết về HTML. Tự trình duyệt sẽ cập nhật các định nghĩa thêm từ CSS vào cho các thẻ HTML đó.

Cú pháp:

```
<head>
<style type="text/css">
    Selector {property1:value 1; property2:value 2; ...; }
    (Lặp lại cho mỗi thẻ có thuộc tính cần định dạng)
</style>
</head>
```

Ví dụ 2-2: Internal style sheet

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Ví dụ 2-2</title>
<STYLE TYPE="text/css">
    H1, H2 { color: limegreen; font-family: Arial; }
</STYLE>
</head>
<body>
    <H1>Đây là thẻ H1 có định dạng bằng css</H1><BR>
    <H2>Đây là thẻ H2 có định dạng bằng css</H2><BR>
    <H3>Đây là thẻ H3 với kiểu mặc định</H3>
</body>
</html>
```



Hình 2-2: Trang Vidu2-2.html

Ở ví dụ này, do ý định của người thiết kế, các thẻ `<h1>` và `<h2>` có cùng định dạng về màu sắc và phông chữ. Do đó, thay vì phải viết hai dòng CSS cho phép ta viết định nghĩa CSS với dạng nhóm, tên các phần tử trong nhóm cách nhau bằng dấu phẩy (,) và sẽ có

cùng định dạng như nhau. Mặc khác, do bản thân các thẻ heading đã được quy định độ lớn riêng nên nếu không định nghĩa lại độ lớn cho các thẻ này bằng CSS thì các thẻ heading sẽ được hiển thị với độ lớn mặc định mà trình duyệt hiểu từ thẻ HTML đó.

**Ưu điểm của CSS loại 2:** CSS loại 2 có độ ưu tiên đứng thứ hai, chỉ sau CSS loại 1. Ưu điểm thứ hai là khi sử dụng CSS loại 2 chúng ta sẽ dễ dàng điều khiển kiểu theo từng tài liệu. Một ưu điểm nữa là không yêu cầu có thêm tập tin lưu trữ thông tin kiểu.

**Nhược điểm:** Do được viết trong phần header của trang Web nên khi muốn có những định dạng CSS tương tự cho trang khác chúng ta cần phải viết lại các định nghĩa đó.

### 2.2.3 External style sheet

External style sheet là một bảng kiểu được lưu trữ thành một file bên ngoài và được liên kết với trang HTML. Do đó, bạn có thể gọi chúng vào bất kỳ trang HTML nào bạn muốn. Nếu bạn đang sử dụng Dreamweaver, để soạn thảo tập tin CSS, bạn vào File → New → Basic Page → CSS. Chúng ta dùng kiểu viết 1 cho bảng kiểu loại 3 này.

Cú pháp:

```
Selector {property1:value 1; property2:value 2; ...; }
```

Sau đó lưu tập tin lại, tập tin sẽ có phần mở rộng là \*.css. Để sử dụng bảng mẫu đã định nghĩa, ta gọi tập tin CSS trong phần `<head>` của trang với thẻ `<link>`.

Cú pháp gọi file:

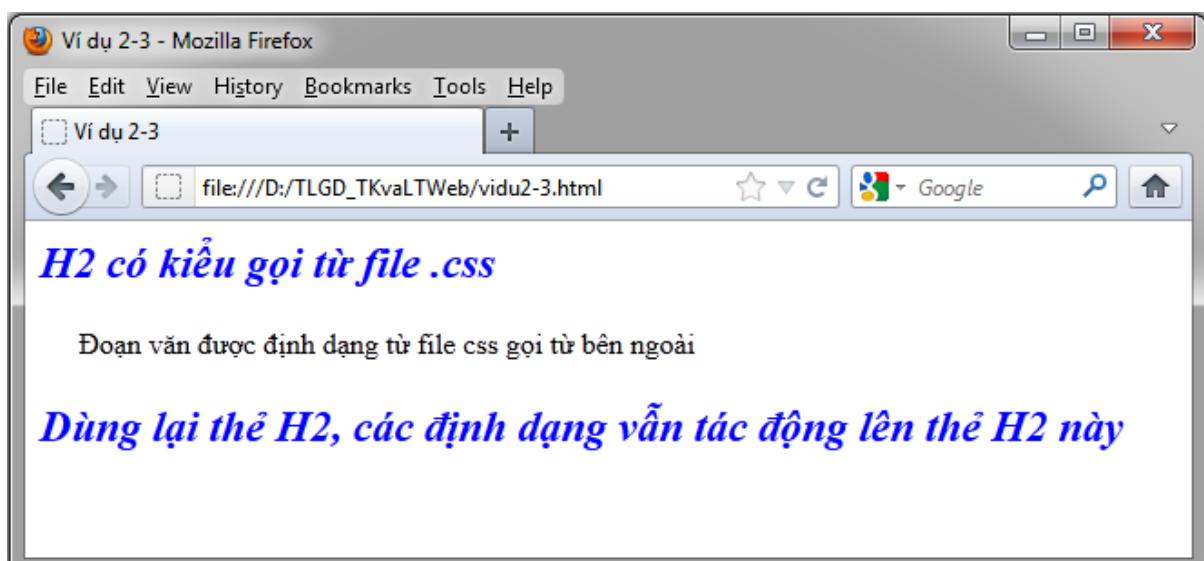
```
<head>
    <link rel="stylesheet" type="text/css" href="URL">
</head>
```

Ví dụ 2-3-1: Tạo trang bangmau.css

```
H2 {
    color:blue;
    font-style:italic;
}
P {
    text-align:justify;
    text-indent:16pt;
    font:"Courier New", Courier, monospace;
}
```

Ví dụ 2-3-2: Tạo trang Vиду2-3.html gọi file CSS vừa tạo

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Ví dụ 2-3</title>
<link rel="stylesheet" type="text/css" href="bangmau.css">
</head>
<body>
    <H2> H2 có kiểu gọi từ file .css</H2>
    <P> Đoạn văn được định dạng từ file css gọi từ bên ngoài
</P>
    <H2>Dùng lại thẻ H2, các định dạng vẫn tác động lên thẻ
H2 này</H2>
</body>
</html>
```



**Hình 2-3: Trang Vиду2-3.html**

Ưu điểm: CSS loại 3 có thể dùng để đặt kiểu cho nhiều tài liệu trong một style sheet, thông tin style được lưu tạm bởi trình duyệt nên sẽ không tốn quá nhiều thời gian cho các lần truy cập sau.

Nhược điểm: Cần thêm thời gian để tải style sheet trong lần truy cập đầu tiên, có thể làm chậm việc hiện trang lên hoặc gây ra hiệu ứng “chớp” nếu style sheet quá nặng. Có thể một số thuộc tính đặt trong CSS loại 3 sẽ không được hiển thị nếu trong tài liệu có dùng CSS loại 1 hoặc 2 để định dạng trước đó.

## 2.3 Vận dụng CSS vào thiết kế giao diện

### 2.3.1 Bộ chọn - Selector

Tùy từng trường hợp mà bộ chọn (Selector) trong kiểu viết 1 có thể là tên thẻ, nhóm thẻ, tên lớp, tên ID,... Chúng ta sẽ tìm hiểu từng trường hợp cụ thể sau đây.

SELECTOR	MÔ TẢ	VÍ DỤ
Element	Element chính là phần tử HTML hay thẻ HTML nhưng chỉ lấy phần tên thẻ mà không có dấu mở thẻ.	<i>H1 {color: red;}</i> => Nếu H1 được gọi thì luôn có màu đỏ
#id	Chọn bất kỳ thẻ HTML nào có thuộc tính id được chỉ định	<i>#test {color:green;}</i> => Bất kỳ thẻ nào có id= "test" đều có màu xanh lá
Element#id	Chỉ thẻ HTML được chỉ định và thẻ đó có id được chỉ định thì mới nhận được định dạng của CSS.	<i>H1#test {color:green;}</i> => Chỉ những <h1 id= "test"> mới có định dạng chữ màu xanh lá
.class	Chọn bất kỳ thẻ HTML nào có giá trị class được chỉ định để nhận định dạng CSS	<i>.note {color: yellow;}</i> => Bất kỳ thẻ nào có class= "note" đều có định dạng là màu vàng
Element.class	Chọn thẻ HTML được định trước với một giá trị class được cài đặt CSS mới nhận được định dạng.	<i>H1.note {text-decoration: underline;}</i> => Chỉ có những thẻ <h1> nào được đặt class= "note" mới có định dạng là chữ được gạch dưới
Grouping	Áp dụng cùng một định dạng kiểu đối với một nhóm thẻ.	<i>h1,h2,h3 {background-color: orange;}</i> => Tất cả các thẻ <h1>, <h2>, <h3> đều có nền chữ màu cam
Contextual	Chọn thẻ con theo đúng thứ tự chỉ định mới nhận được định dạng CSS này	<i>p strong {color:purple;}</i> => Chỉ khi đúng thứ tự <p><strong> nội dung </strong></p> thì nội dung mới có định dạng màu tím

### 2.3.2 Các thuộc tính CSS thông dụng trong thiết kế giao diện

ĐỊNH DẠNG	THUỘC TÍNH CSS	Ý NGHĨA
Văn bản	<code>color:red;</code>	Màu chữ (dùng từ tiếng Anh)
	<code>color:#990000;</code>	Màu chữ (dùng mã màu)
	<code>background:grey;</code>	Màu nền (dùng từ tiếng Anh)
	<code>background:#999999;</code>	Màu nền (dùng mã màu)
	<code>word-spacing:n;</code>	Khoảng cách giữa các từ, với n là số pixel mà bạn nhập vào
	<code>letter-spacing:n;</code>	Khoảng cách giữa các ký tự, với n là số pixel mà bạn nhập vào
	<code>text-align: left / right / center / justify;</code>	Canh lề văn bản (trái/phải/giữa/đều 2 bên)
	<code>text-transform: capitalize / uppercase / lowercase;</code>	Chuyển thành dạng chữ: viết hoa đầu mỗi từ / toàn chữ in hoa / toàn chữ in thường
	<code>font-family:Times New Roman, Times, serif;</code> <code>font: Arial, Helvetica, sans-serif;</code> <code>font:Courier New, Courier, monospace;</code> ...	Chọn kiểu chữ (font); nếu máy người dùng không có font được chỉ định thì trình duyệt sẽ lấy font mặc định cùng họ với font đã chọn để hiển thị
	<code>font-size: xx-small / x-small / small / medium / large / x-large / xx-large;</code> <code>font-size:12pt;</code>	Định cỡ chữ (rất rất nhỏ / rất nhỏ / nhỏ / trung bình / lớn / rất lớn / rất rất lớn) Định cỡ chữ (giá trị cụ thể từ 9px đến 36px)
Ký hiệu đầu dòng danh sách	<code>font-style: italic;</code>	Tạo chữ nghiêng
	<code>font-weight: bold;</code>	Tạo chữ đậm
	<code>fext-decoration: underline;</code>	Gạch dưới văn bản
	<code>list-style: none;</code>	Không dùng Bullet
	<code>list-style: circle;</code>	Chấm tròn rỗng O
	<code>list-style: disc;</code>	Chấm tròn đen ●

ĐỊNH DẠNG	THUỘC TÍNH CSS	Ý NGHĨA
Nền	<i>list-style: square;</i>	Chấm đen vuông ■
	<i>list-style: decimal;</i>	Đánh số Ả Rập (1, 2, 3, ...)
	<i>list-style: lower-alpha;</i>	Thứ tự alpha thường (a, b, c, ...)
	<i>list-style: upper-alpha;</i>	Thứ tự alpha hoa (A, B, C, ...)
	<i>list-style: upper-roman;</i>	Số La Mã hoa (I, II, III, ...)
	<i>list-style: lower-roman;</i>	Số La Mã thường (i, ii, iii, ...)
	<i>list-style-image: url;</i>	Dùng hình làm Bullet
Định dạng đường viền	<i>background-color:màu / tên màu;</i>	Màu nền
	<i>background-image:đường dẫn đến ảnh nền;</i>	Ảnh nền
	<i>background-position: left / right / center / top / bottom / inherit;</i>	Vị trí đặt ảnh nền: trái/phải/giữa/trên/dưới/ké thừa
	<i>background-repeat: repeat / repeat-x / repeat-y / no-repeat;</i>	Cách lặp ảnh: lặp cho cả trang/lặp theo chiều ngang / lặp theo chiều đứng / không lặp
	<i>background-attachment: scroll / fixed;</i>	Giữ ảnh cuộn /không cuộn theo trang của trình duyệt
	<i>border-style:</i>	Kiểu đường viền
	<i>border-collapse: collapse;</i>	Đường viền lún
	<i>border-bottom-style:</i>	Kiểu đường viền phía dưới
	<i>border-left-style:</i>	Kiểu đường viền bên trái
	<i>border-right-style:</i>	Kiểu đường viền bên phải

ĐỊNH DẠNG	THUỘC TÍNH CSS	Ý NGHĨA
	<i>border-top-style:</i>	Kiểu đường viền phía trên
	<i>border-left-color:</i>	Màu đường viền bên trái
Hypertext link	<i>text-decoration: none</i>	Không gạch dưới
	<i>a:visited{color: #rrggbb;}</i>	Định dạng cho link khi đã được click
	<i>a:link{color:#rrggbb;}</i>	Định dạng cho link chưa được click
	<i>a:active{color:#rrggbb; }</i>	Định dạng cho những link đang xem
	<i>a:hover{color:#rrggbb;}</i>	Định dạng khi trỏ lướt qua link => Bạn có thể thay đổi màu, phóng to hoặc thu nhỏ chữ hoặc thêm nhiều hiệu ứng khác nữa.
Định vị	<i>float: left/right/none;</i>	Float (theo như nghĩa tiếng Việt là thả trôi) đây là thuộc tính cố định một thành phần web về bên trái hay bên phải không gian bao quanh nó. Mặc định là <i>none</i> . Đây là thuộc tính quan trọng để dàn trang trên Web, đặc biệt là định vị cho các Layout dùng <i>&lt;div&gt;</i>
	<i>left</i>	Đây là một thuộc tính dịch đối tượng từ bên trái với số pixel chỉ định sau dấu hai chấm của thuộc tính <i>left</i> .
	<i>right</i>	Đây là một thuộc tính dịch đối tượng từ bên phải với số pixel chỉ định sau dấu hai chấm của thuộc tính <i>right</i> .

ĐỊNH DẠNG	THUỘC TÍNH CSS	Ý NGHĨA
	<i>top</i>	Đây là một thuộc tính dịch đối tượng từ phía trên với số pixel chỉ định sau dấu hai chấm của thuộc tính <i>top</i> .
	<i>bottom</i>	Đây là một thuộc tính dịch đối tượng từ phía dưới với số pixel chỉ định sau dấu hai chấm của thuộc tính <i>bottom</i> .
	<i>position: absolute;</i>	<p>Đây là một thuộc tính định vị tuyệt đối, có nghĩa là các thành phần được định vị không để lại bát cứ một khoảng trống nào trong tài liệu. Thông thường người ta kết hợp với các thuộc <i>left/right/top/bottom</i> để xác định tọa độ đặt đối tượng.</p> <p>Trong kỹ thuật tạo menu động với CSS, bạn cần cố định menu cha với vị trí tuyệt đối</p>
	<i>position: relative;</i>	<p>Đây là một thuộc tính định vị tương đối cho một thành phần, là sự định vị được tính từ vị trí gốc trong tài liệu. Các thành phần đã được định vị tương đối sẽ để lại khoảng không trong tài liệu.</p> <p>Trong kỹ thuật tạo menu động với CSS, bạn cần định vị cho menu con là tương đối, tức là định vị được tính từ vị trí của menu cha.</p>

ĐỊNH DẠNG	THUỘC TÍNH CSS	Ý NGHĨA
	<i>margin</i>	<p>Định khoảng cách từ viền của đối tượng đến một đối tượng khác hoặc đến một trang. Giá trị bên trong margin có thể là 4 giá trị xác định khoảng cách tương ứng theo thứ tự: trên, phải, dưới, trái. Ví dụ: <i>margin: 2px 5px 4px 7px;</i> Có nghĩa là từ viền đối tượng cách viền trang hoặc đối tượng khác về hướng trên là 2px, hướng phải là 5px, hướng dưới là 4px và hướng trái là 7px.</p> <p>Giá trị bên trong margin cũng có thể có 2 giá trị. Khi đó khoảng cách sẽ được xác định tương ứng theo cặp: trên và dưới, phải và trái. Ví dụ: <i>margin: 2px 4px;</i> Có nghĩa là từ viền của đối tượng cách đối tượng khác hoặc viền trang theo hướng trên và dưới đều là 2px, theo hướng phải và trái đều là 4px.</p> <p>Cũng có thể giá trị bên trong margin chỉ có 1 giá trị. Khi đó khoảng cách sẽ được xác định theo bộ tứ, tức là khoảng cách về 4 hướng của đối tượng so với đối tượng khác hoặc viền trang đều như nhau. Ví dụ: <i>margin: 4px;</i> Có nghĩa là từ viền của đối tượng đến viền trang hoặc đối tượng khác theo hướng trên là 4px, theo hướng phải cũng là 4px, theo hướng dưới cũng là 4px và theo hướng trái cũng là 4px.</p>

ĐỊNH DẠNG	THUỘC TÍNH CSS	Ý NGHĨA
	<i>padding</i>	Định khoảng cách từ nội dung đặt trong đối tượng đến viền của đối tượng đó (hay còn gọi là vùng đệm). Các giá trị thuộc tính cũng tương tự như margin.

Ví dụ 2-4-1: Chuẩn bị giao diện làm menu.

Mở lại ví dụ 1-2-6: danh sách không thứ tự và lưu lại với tên mới là **Vidu2-4.html**. Sau đó thêm vào một số thẻ được in đậm bên dưới:

```

-----+
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Ví dụ 1-2-6</title>
<link rel="stylesheet" type="text/css" href="Vidu3-4.css">
</head>
<body>
    <h1>Các nội dung trong môn học</h1>
    <div id="menu">
        <ul type="square">
            <li><a href="#">HTML</a>
                <ul type="circle">
                    <li><a href="#">Các thẻ HTML cơ bản</a></li>
                    <li><a href="#">Hyperlink và Frame</a></li>
                    <li><a href="#">Table</a></li>
                    <li><a href="#">Form và các Controls</a> </li>
                </ul>
            </li>
            <li><a href="#">CSS</a>
                <ul type="disc">
                    <li><a href="#">Khái niệm</a></li>
                    <li><a href="#">Phân loại</a></li>
                    <li><a href="#">Thiết kế giao diện bằng
CSS</a></li>
                </ul>
            </li>
            <li><a href="#">Javascript</a></li>
        </ul>
    </div>
</body>
</html>
-----+

```

Một số lưu ý khi viết mã nguồn của trang **Vidu2-4.html**, như đã trình bày trước đây trong ví dụ 1-2-6, cần đảm bảo cấp độ của menu, các phần tử *<li>* đóng vai trò là cha phải thật sự chứa *<ul>* cấp con của nó. Bên cạnh đó, các liên kết phải được đặt trong phần tử *<li>* và do chưa có liên kết đến các trang khác nên các liên kết tạm thời ta đặt dấu # trong *href*.

Ví dụ 2-4-2: Vận dụng CSS làm menu động và lưu vào tập tin có tên **Vidu2-4.css**

MÃ NGUỒN	GIẢI THÍCH
<pre>*{     margin:0px;     padding:0px; }  div#menu{     border:1px solid #030;     float:left;     background-color:#690;     width:500px;     height:90px; }  div#menu ul{     float:left;     list-style:none; }  div#menu ul li{     float:left;     position:relative; }  div#menu ul li a{     text-decoration:none;     color:#FFF;     padding:5px 30px;     display:block; }</pre>	<p>Reset tất cả các khoảng cách mặc định của trình duyệt về 0px để đảm bảo các trình duyệt hiển thị menu là như nhau.</p> <p>Định dạng cho thẻ <i>&lt;div id="menu"&gt;</i>:</p> <ul style="list-style-type: none"> <li>- Hiển thị đường viền dạng solid, bề dày 1px và màu viền #030.</li> <li>- Thả trôi thẻ <i>&lt;div&gt;</i> về bên trái.</li> <li>- Khung chứa rộng 500px, cao 90px và có màu nền là #690.</li> </ul> <p>Định dạng cho <i>&lt;ul&gt;</i> nằm trong <i>&lt;div id="menu"&gt;</i>:</p> <ul style="list-style-type: none"> <li>- Thả trôi menu về bên trái và bỏ ký hiệu đầu danh sách.</li> </ul> <p>Định dạng cho <i>&lt;li&gt;</i> trong <i>&lt;ul&gt;</i> cấp 1.</p> <ul style="list-style-type: none"> <li>- Thả trôi các phần tử trong menu về bên trái để hiển thị chúng nằm ngang nhau.</li> <li>- Định vị cho menu cha là tương đối.</li> </ul> <p>Định dạng cho các thẻ <i>&lt;a&gt;</i> trong <i>&lt;li&gt;</i>:</p> <ul style="list-style-type: none"> <li>- Bỏ đường gạch chân của liên kết.</li> <li>- Chữ của liên kết màu trắng.</li> <li>- Lê trên và dưới của liên kết cách nội dung 5px. Tương tự lề trái, phải là 30px.</li> <li>- Cho liên kết được bao khói trong <i>&lt;li&gt;</i>.</li> </ul>

```

div#menu ul li a:hover{
    color:#690;
    font-weight:bold;
    text-decoration:underline;
    background-color:#FFF;
}

div#menu ul li ul{
    display:none;
    position:absolute;
    width:500px;
}

div#menu ul li:hover ul{
    display:block;
}

```

Khi di chuyển chuột qua liên kết:

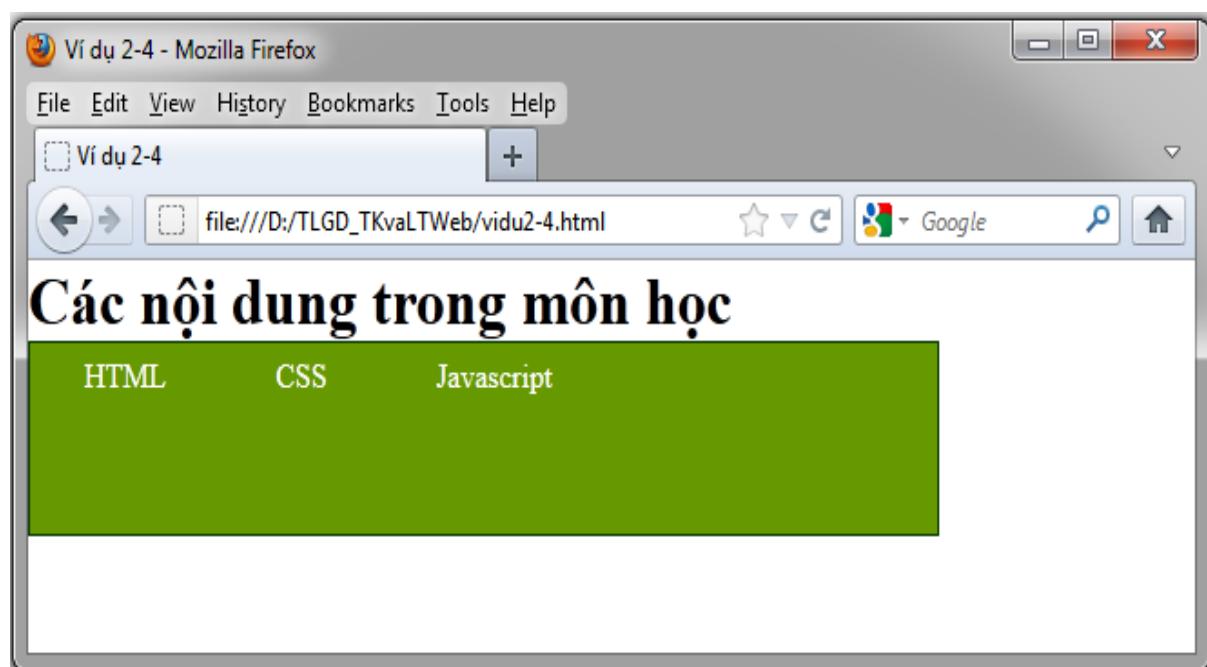
- Đổi màu chữ của liên kết.
- In đậm, cho xuất hiện lại đường gạch chân, xuất hiện màu nền trắng.

Định nghĩa CSS cho menu con:

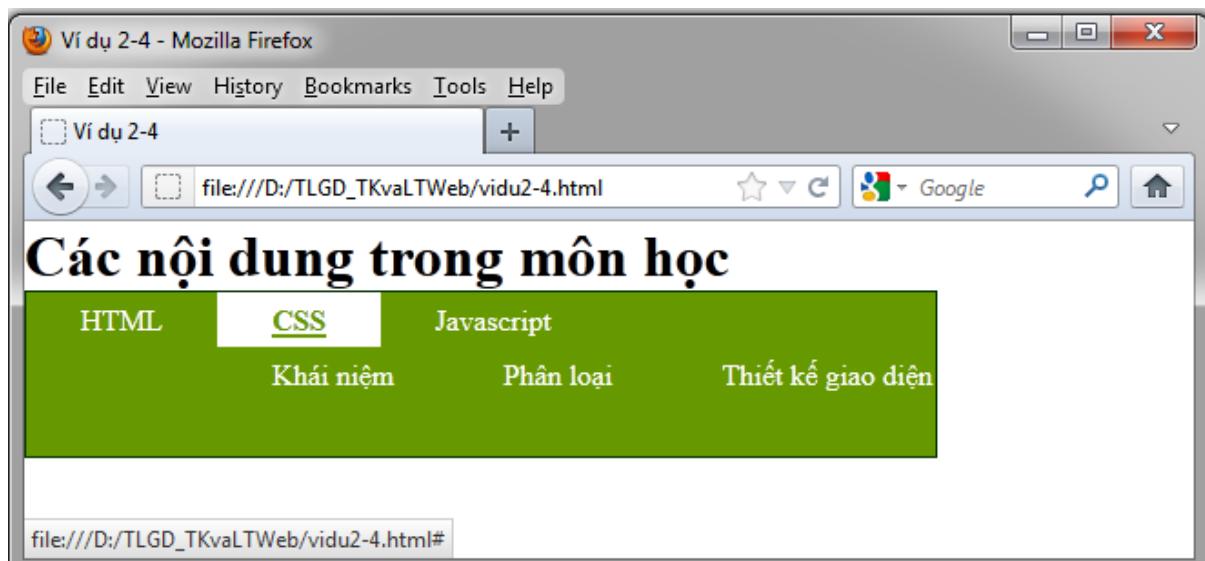
- Không cho menu con xuất hiện.
- Vị trí của menu con là tuyệt đối theo vị trí của menu cha.
- Độ rộng của menu con là 500px.

Khi di chuyển chuột vào menu cha sẽ có tác dụng lên menu con:

- Cho hiện lại menu con.



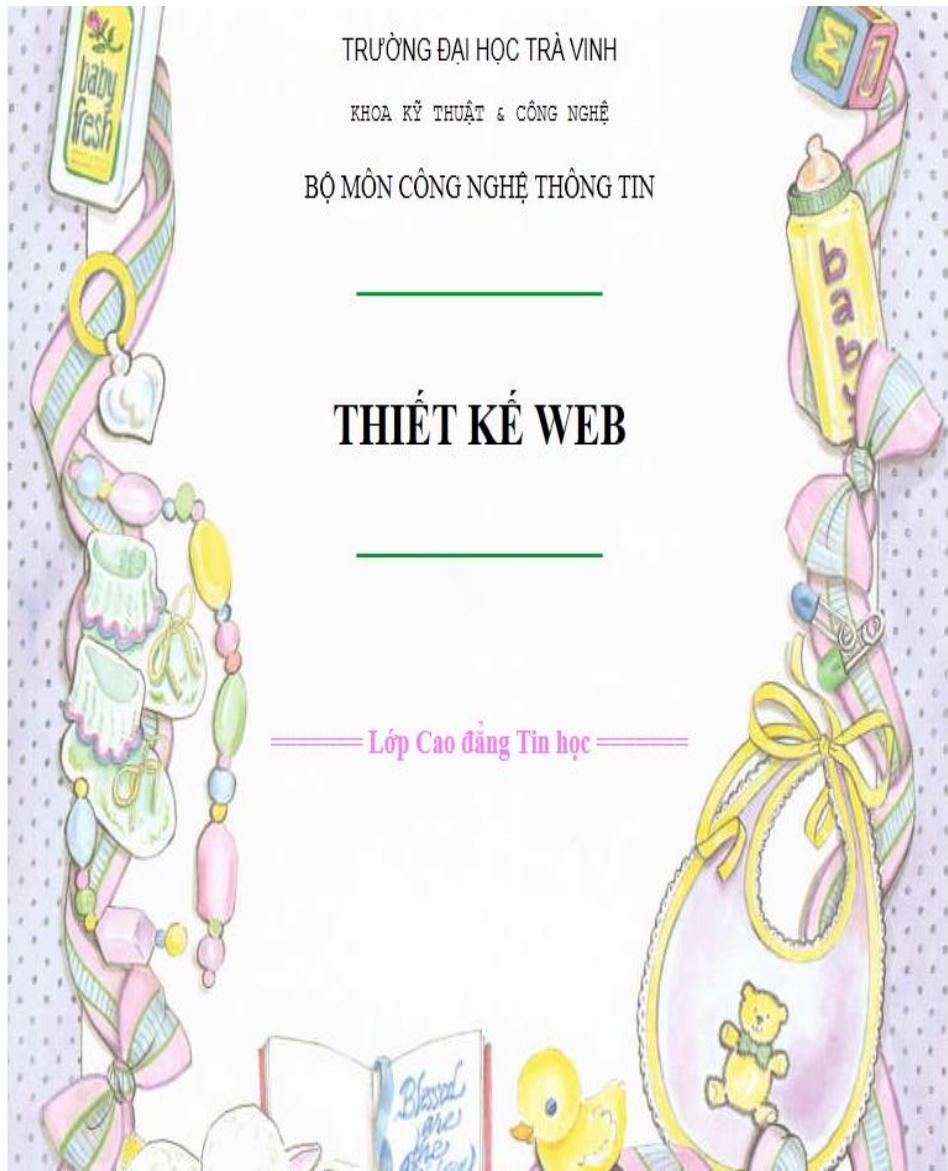
**Hình 2-4: Trang Vиду2-4.html**



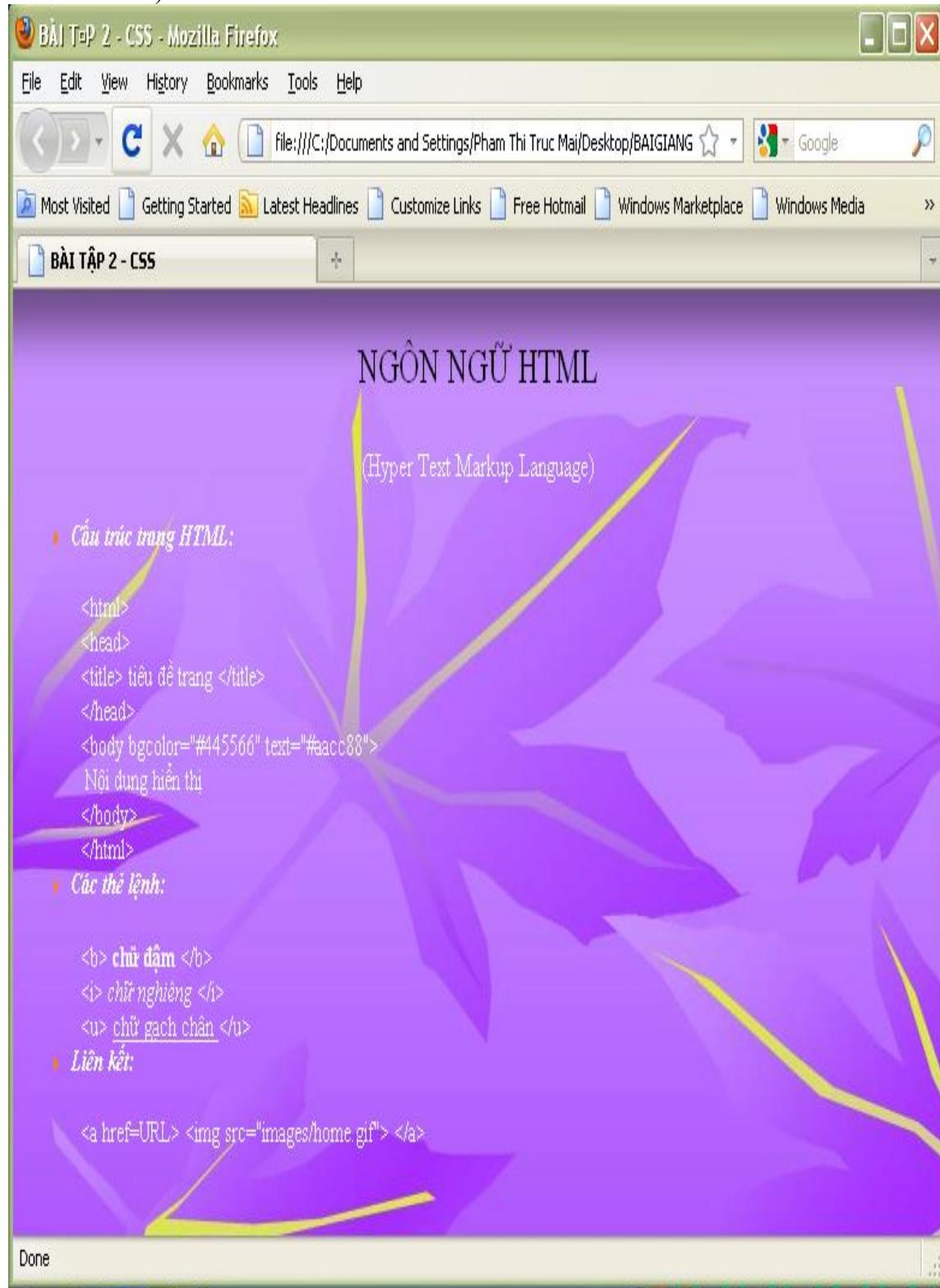
Hình 2-5: Trang Vidu2-4.html khi đưa con trỏ qua liên kết CSS

❖ Câu hỏi (bài tập) củng cố:

1. Dùng css – bảng kiểu trực tiếp để thiết kế lại trang **baitap1.html** sao cho code HTML trở nên đơn giản nhất và lưu lại với tên mới là **CSS\_baitap1.html**

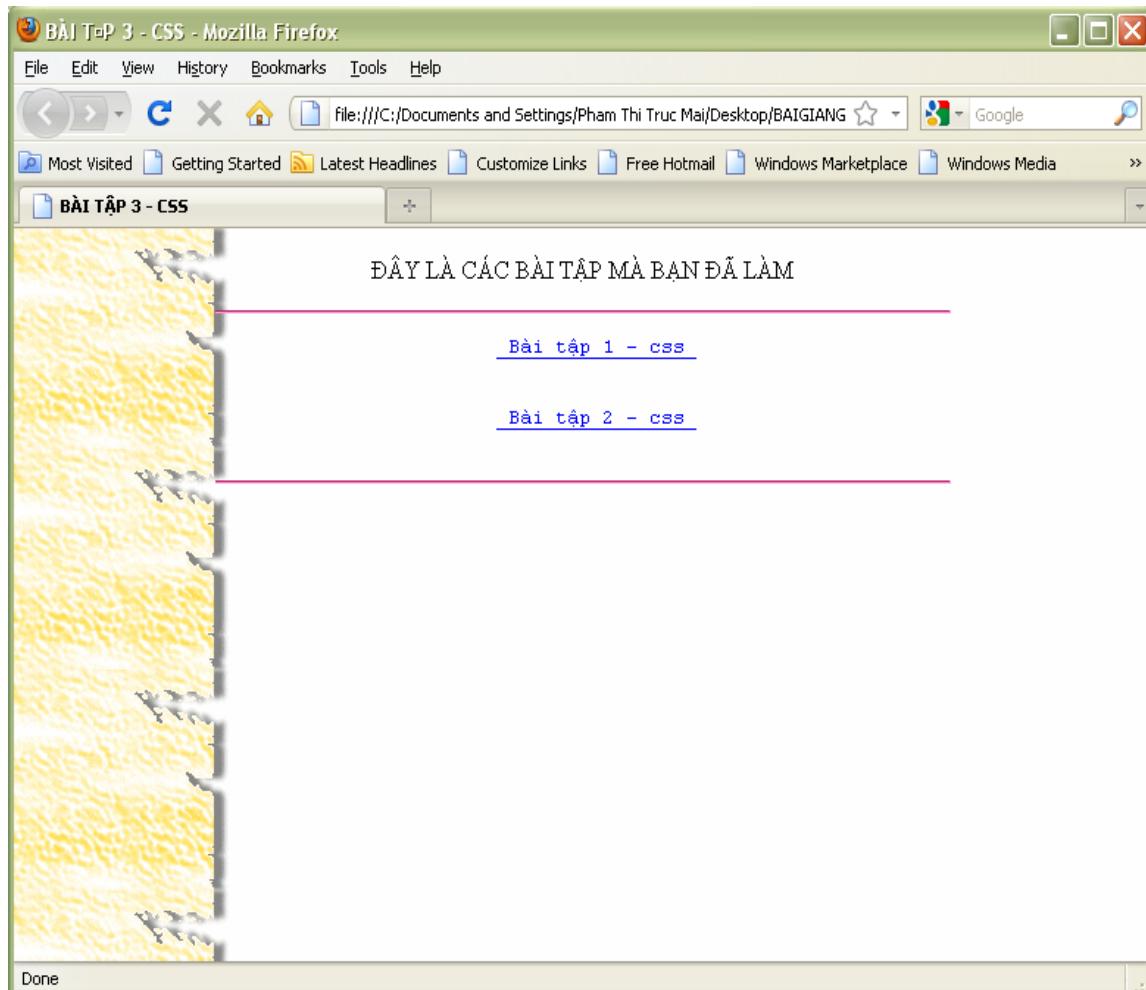


2. Sử dụng css – bảng kiểu nhúng trong tài liệu để thiết kế lại trang **baitap2.html** sao cho code HTML trở nên đơn giản nhất và lưu lại với tên mới là **CSS\_baitap2.html** (chú ý thay đổi ký hiệu của danh sách không thứ tự thành dạng hình ảnh)



3. Sử dụng cách gọi css từ một file bên ngoài, thiết kế trang Web có nội dung như sau. Lưu tập tin CSS với tên là CSS\_baitap3.css và lưu tập tin HTML với tên CSS\_baitap3.html

Chú ý: nếu link chưa mở lần nào thì có màu như trong hình. Nếu rê chuột vào link, chữ sẽ đổi màu và có font-size lớn hơn. Nếu link đã vị click ít nhất 1 lần thì sẽ có màu vàng cam. Nếu là link đang được mở thì đường link sẽ đậm lên.



4. Hãy vận dụng 1 trong 3 loại CSS để thực hành tạo menu động dạng DropDown như sau và lưu bài tập với tên CSS\_baitap4.html

Trang chủ	HTML	CSS	JAVASCRIPT	PHP
	Các ký hiệu đặc biệt			
	Các thẻ định dạng văn bản			
	Các thẻ danh sách và liên kết			
	Thẻ tạo bảng			
	Form và các control			
	Khung			

Cấu trúc menu:

- [Trang chủ](#)
- [HTML](#)
  - [Các ký hiệu đặc biệt](#)
  - [Các thẻ định dạng văn bản](#)
  - [Các thẻ danh sách và liên kết](#)
  - [Thẻ tạo bảng](#)
  - [Form và các control](#)
  - [Khung](#)
- [CSS](#)
  - [Định dạng nền](#)
  - [Định dạng chữ](#)
  - [Làm DropDown Menu](#)
  - [Làm FlyOut Menu](#)
- [JAVASCRIPT](#)
- [PHP](#)

5. Hãy vận dụng 1 trong 3 loại CSS để thực hành tạo menu động dạng FlyOut như sau và lưu bài tập với tên CSS\_baitap5.html

Menu chính	
Trang chủ	
Tác giả	
Tác phẩm	Truyện ngắn
Liên hệ	Truyện dài
	Truyện cười
	Thơ

Cấu trúc menu:

## Menu chính

- [Trang chủ](#)
- [Tác giả](#)
  - [Tác giả A](#)
  - [Tác giả B](#)
  - [Tác giả C](#)
- [Tác phẩm](#)
  - [Truyện ngắn](#)
  - [Truyện dài](#)
  - [Truyện cười](#)
  - [Thơ](#)
- [Liên hệ](#)

6. Bài tập nâng cao: Hãy dùng thẻ `<div>` và vận dụng các thuộc tính CSS đã được học để thiết kế giao diện cho bài tập nâng cao 1, 2 ở bài 3 chương 1 theo các dạng Layout sau:

Layout 1:

Banner		
Menu chính	Nội dung các trang	Menu phụ
Footer		

Layout 2:

Menu chính	Banner
	Nội dung các trang
Menu phụ	
Footer	

## CHƯƠNG 3

### NGÔN NGỮ JAVASCRIPT

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Vận dụng JavaScript hỗ trợ thiết kế web

#### 3.1 Giới thiệu

Khi duyệt trang Web, nếu trang Web cho phép người dùng nhập liệu, bạn cần viết một phương thức để kiểm soát dữ liệu nhập có hợp lệ hay không. Để thực hiện điều này, bạn có thể sử dụng một trong hai phương thức: kiểm tra tại trình duyệt của người dùng trước khi gửi dữ liệu về Server (dùng các Client Script như JavaScript, VBScript,...), gửi dữ liệu về máy chủ rồi mới kiểm tra và xử lý dữ liệu (dùng các Server Script như PHP chẳng hạn).

Nếu dữ liệu kiểm tra phía Server không hợp lệ, có nghĩa là người dùng dữ liệu không hợp lệ bạn cũng đã cho phép họ gửi dữ liệu lên Server. Sau đó, nếu dữ liệu không hợp lệ, trang Web sẽ chuyển về trang ban đầu, bạn cần lưu giữ lại các giá trị trước khi submit nhằm trình bày lại thông tin mà người dùng nhập trước đó. Tuy nhiên, với trường hợp này người dùng đã submit, trang Web này đã mất dữ liệu trước đó, khi trở lại bạn cần thông báo cho người dùng biết những giá trị nào không hợp lệ. Làm như vậy bạn phải xử lý rất nhiều vấn đề.

Một giải pháp cho đến thời điểm bây giờ vẫn còn hiệu quả là sử dụng Client Script, nhằm kiểm tra dữ liệu nhập của người dùng trước khi submit lên Server. Đây là trường hợp điển hình bạn cần sử dụng Client Script. Ngoài ra còn nhiều chức năng khác mà Client Script đóng vai trò như một phần quan trọng trong thiết kế cũng như lập trình Web.

JavaScript là ngôn ngữ thông dịch. Để sử dụng ngôn ngữ JavaScript, bạn cần phải tuân thủ một số quy tắc như:

Trong JavaScript có sự phân biệt CHỮ HOA / chữ thường.

Một câu lệnh đơn kết thúc bằng dấu chấm phẩy (;).

Tập hợp nhiều câu lệnh đơn được đặt trong cặp dấu { } được gọi là một khái

lệnh.

Ghi chú một dòng: //

Ghi chú nhiều dòng: /\* \*/

Dữ liệu là chuỗi, phải được đặt trong cặp dấu nháy kép ( “ ” ).

Có thể dùng dấu + để nối các chuỗi hoặc các biến.

Có thể đặt JavaScript ở bất cứ đâu trong tập tin HTML.

Có thể tạo nhiều Script trong cùng một tập tin HTML.

Có thể chèn thẻ HTML vào JavaScript (đặt trong nháy kép “ ” như một chuỗi).

Có hai cú pháp để gọi JavaScript:

Nhúng trực tiếp vào trang HTML:

```
<script language="JavaScript">
    //Các câu lệnh JavaScript;
</script>
```

Gọi file từ bên ngoài:

```
<script language="javascript" src="URL "></script>
```

Trong đó, URL là đường dẫn đến file JavaScript. Tập tin này có phần mở rộng là \*.js.

## 3.2 Biến, toán tử, toán hạng, hằng, hàm, lệnh trong JavaScript

### 3.2.1 Biến trong JavaScript

Biến trong JavaScript không cần xác định kiểu trước, kiểu của dữ liệu sẽ được xác định tại thời điểm gán giá trị cho biến. Một số lưu ý khi đặt tên biến là tên biến phải bắt đầu bằng ký tự chữ cái hoặc ký tự \_, không dùng các ký tự đặc biệt để đặt tên biến: ( , [ , { , # , &, ..., không bắt đầu bằng ký số, không dùng khoảng trắng, không được trùng với các từ khóa trong JavaScript.

Cách khai báo:

```
var tên_bien;
```

Hoặc:

```
ten_bien=gia_trị;
```

Danh sách các từ khóa trong JavaScript:

abstract	extends	int	super
boolean	false	interface	switch
break	final	long	synchronized
byte	finally	native	this
case	float	New	throw
catch	for	Null	throws
char	function	package	transient
class	goto	private	true
const	if	protected	try
continue	implements	public	var
default	import	return	val
do	in	short	while
double	instanceof	static	with
else			

Biến trong JavaScript cũng phân làm 2 loại: Biến toàn cục và biến cục bộ. Biến toàn

cục được khai báo ngoài các hàm và có phạm vi hoạt động từ vị trí khai báo trở về sau trong chương trình. Biến cục bộ được khai báo trong chương trình con hoặc bên trong hàm và chỉ có phạm vi hoạt động từ vị trí khai báo đến kết thúc chương trình con hoặc kết thúc hàm.

### 3.2.2 Toán tử

Toán tử số học:

TOÁN TỬ	CHỨC NĂNG	VÍ DỤ	KẾT QUẢ
+	Cộng	$x=2; x+2$	4
-	Trừ	$x=2; 5-x$	3
*	Nhân	$x=4; x*5$	20
/	Chia	$5/2$	2.5
%	Lấy dư	$5\%2$	1
++	Tăng 1	$x=5; x++$	6
--	Giảm 1	$x=5; x--$	4

Toán tử gán:

TOÁN TỬ	VÍ DỤ	TƯƠNG ĐƯƠNG
=	$x = y$	$x = y$
+=	$x += y$	$x = x+y$
-=	$x -= y$	$x = x-y$
*=	$x *= y$	$x = x*y$
/=	$x /= y$	$x = x/y$
%=	$x \%= y$	$x = x \% y$

Toán tử so sánh: Trả về True nếu đúng, trả về False nếu sai:

TOÁN TỬ	CHỨC NĂNG	VÍ DỤ	KẾT QUẢ
==	Bằng	$5==8$	return false
!=	Không bằng	$5!=8$	return true
>	Lớn hơn	$5>8$	return false
<	Nhỏ hơn	$5<8$	return true
>=	Lớn hơn hoặc bằng	$5>=8$	return false
<=	Nhỏ hơn hoặc bằng	$5<=8$	return true

Toán tử logic: Trả về True nếu đúng, trả về False nếu sai:

TOÁN TỬ	CHỨC NĂNG	VÍ DỤ	KẾT QUẢ
&&	Và	x = 6; y = 3 ; (x < 10 && y > 1)	return true
	Hoặc	x = 6 ; y = 3; (x==5    y==5)	return false
!	Not	X = 6; y = 3; !(x==y)	return true

Toán tử điều kiện:

```
(Điều kiện) ? value1: value2;
```

Nếu điều kiện đưa vào trong dấu () là đúng thì trả về value1. Ngược lại thì trả về value2.

### 3.2.3 Hằng số

Khác với các ngôn ngữ lập trình khác, JavaScript không có kiểu hằng số CONST. Giá trị của biến sẽ được xác định tại thời điểm gán biến cho đến khi biến được gán lại với giá trị mới.

### 3.2.4 Hàm

JavaScript hỗ trợ các hàm có sẵn hoặc bạn có thể tự định nghĩa một hàm mới. Các hàm có sẵn thông dụng:

Để xuất dữ liệu lên Web, bạn có thể sử dụng các hàm write() hoặc writeln(). Trong đó hàm writeln() sẽ kèm theo ký tự xuống hàng khi kết thúc chuỗi hiển thị.

```
document.write("Nội dung cần hiển thị");
document.writeln("Nội dung cần hiển thị");
```

Hiển thị hộp thoại: có 3 dạng hộp thoại tùy theo dung ý thông báo mà bạn có thể chọn một trong ba loại này. Thứ nhất là hộp thoại OK, trên hộp thoại chỉ có duy nhất một nút OK kèm thông báo. Loại hộp thoại này thường dùng để cảnh báo cho người dùng. Thứ hai là hộp loại YES/NO, trên hộp thoại có hai nút OK và Cancel. Loại hộp thoại này là dạng dành cho người dùng xác nhận thông tin, nếu đồng ý thì chọn OK, nếu không đồng ý thì chọn Cancel. Thứ ba là hộp thoại cho phép người dùng nhập giá trị vào. Trên hộp thoại dạng này

có một khung nhập liệu, nút OK và Cancel. Đây là một dạng hộp thoại nhận thêm thông tin từ phía người dùng. Để xác nhận thông tin bổ sung, người dùng nhấn vào nút OK, nếu nhấn nút Cancel thì bỏ qua.

```
alert ("chuỗi sẽ hiển thị trong hộp thoại ");
```



**Hình 3-1: Hộp thoại OK**

```
confirm ("Bạn có chắc muốn thay đổi giá trị này?");
```



**Hình 3-2: Hộp thoại YES/NO**

```
prompt("Mời bạn nhập thông tin:");
```



**Hình 3-3: Hộp thoại lấy thông tin**

Các hàm xử lý chuỗi gồm hàm: để biến chuỗi (dạng biểu thức) thành biểu thức có thể tính toán được (hoặc trả về giá trị số của một chuỗi số) ta dùng hàm *eval(Chuỗi/biểu thức)*. Trả về trị số nguyên của chuỗi số, nếu không có giá trị, hàm sẽ trả về NaN:

`parseInt("chuỗi số")`. Ví dụ: nếu dùng `parseInt("125.6a")`; thì kết quả trả về sẽ là 125. Nếu dùng `parseInt("15ab6")`; thì kết quả trả về là 15 do 15 là hai ký số đầu tiên trong chuỗi, hàm sẽ hiểu có một phần số trong chuỗi và trả về số mà nó nhận diện được. Còn nếu dùng `parseInt("dh154")`; thì kết quả trả về là NaN do chuỗi đưa vào hàm parseInt lúc này có các ký tự đầu tiên không phải số, hàm sẽ dừng việc kiểm tra và thông báo đây không phải số. Tương tự, chúng ta có hàm trả về trị số thực của chuỗi số, nếu không có giá trị, hàm sẽ trả về NaN: `parseFloat("chuỗi số")`. Nguyên tắc hoạt động của hàm này tương tự như `parseInt` chỉ khác ở chỗ hàm `parseFloat` sẽ nhìn thấy dấu chấm động của số thực. Khi muốn kiểm tra nội dung người dùng nhập vào là chuỗi hay không, bạn dùng hàm `isNaN(giá trị kiểm tra)`. Nếu `giá trị kiểm tra` không phải là số thì trả về True, nếu là số trả về False.

Ngoài ra, chúng ta có thể tự định nghĩa một hàm với cú pháp sau:

```
function tên_hàm ( thamsol, thamso2,... )
{
    //Khai báo các biến sử dụng trong hàm;
    //Các câu lệnh xử lý trong JavaScript;
    [return [giá trị /biểu thức] ];
}
```

Ví dụ 3-1: Hàm tự định nghĩa

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Ví dụ 3-1</title>
<script language="javascript">
    function phepnhan(a,b)
        {x=a*b;
        return x;
        }
</script>
</head>
<body>
    <script language="javascript">
        document.write(phepnhan(2,6)); //gọi hàm và xuất kết quả
    </script>
</body>
</html>
```

Lưu ý: tên hàm tự định nghĩa không được trùng với từ khóa, tên hàm trong JavaScript và không được trùng với tên của các thành phần trong form HTML nếu đang truy xuất dữ

liệu của form.

### 3.2.5 Các lệnh điều khiển trong JavaScript

Câu lệnh điều kiện if / if ... else

Câu lệnh đơn giản, có 2 cú pháp:

Cú pháp 1:

```
if (<Biểu thức điều kiện>)
    {//Khởi lệnh;}
```

Cú pháp này dùng cho trường hợp chỉ cần kiểm tra điều kiện, nếu thỏa điều kiện thì thực hiện công việc chỉ định trong khối lệnh. Nếu không thỏa điều kiện thì không làm gì cả.

Cú pháp 2:

```
if (<Biểu thức điều kiện>
    {//Khởi lệnh 1;}
else
    {//Khởi lệnh 2;}
```

Cú pháp 2 áp dụng cho trường hợp nếu thỏa điều kiện thì thực hiện khối lệnh thứ nhất. Ngược lại nếu không thỏa điều kiện trong if sẽ thực hiện khối lệnh thứ 2.

Nếu có nhiều hơn 2 điều kiện, chúng ta có thể áp dụng các dòng if ... else lồng nhau để có được cấu trúc điều kiện phức tạp

Cú pháp 3:

```
if(<bìểu thức điều kiện1>
    {//Khởi lệnh 1;}
else
    if (<bìểu thức điều kiện 2>)
        {//Khởi lệnh 2;}
    ...
else
    {//Khởi lệnh 3;}
```

Ví dụ 3-2: Một đoạn lệnh kiểm tra các dạng tam giác được viết bằng JavaScript

```

<script language="javascript">
    a=eval(prompt("Nhập cạnh a"));
    b=eval(prompt("Nhập cạnh b"));
    c=eval(prompt("Nhập cạnh c"));
    if(a+b<c || b+c<a || c+a<b)
        alert("Không phải tam giác");
    else
        if(a==b&&b==c&&c==a)
            alert("Tam giác đều");
        else
            if(a==b || b==c || c==a)
                alert("Tam giác cân");
            else
                alert("Tam giác thường");
</script>

```

#### Lựa chọn switch ... case

Trường hợp có nhiều điều kiện, nếu phải dùng cấu trúc if ... else lồng nhau sẽ làm cho code trở nên rườm rà, phức tạp. Cho nên, bạn có thể sử dụng cấu trúc lựa chọn switch ... case cho các trường hợp như thế.

#### Cú pháp 1:

```

switch(Biểu thức)
{
    case value1:
        //Khởi lệnh 1;
        break;
    case value2:
        //Khởi lệnh 2 ;
        break;
    .....
    case valuek:
        //Khởi lệnh k ;
        break;
}

```

Với cú pháp 1, nếu biểu thức kiểm tra trong switch không thỏa mang bất kỳ giá trị nào trong các case, khi đó switch sẽ không trả kết quả.

#### Cú pháp 2:

```

switch(Biểu thức)
{
    case value1:
        //Khởi lệnh 1;
        break;
    case value2:
        //Khởi lệnh 2 ;
        break;
    .....
    case valuek:
        //Khởi lệnh k ;
        break;
    default :
        //Khởi lệnh k+1;
        break;
}

```

Nếu dùng cú pháp 2, khi giá trị của biểu thức không trùng với các giá trị liệt kê sẵn trong case thì switch sẽ thực hiện khởi lệnh đặt trong default.

Ví dụ 3-3: dùng switch ... case để giải bài toán xác định số ngày trong tháng

```

<script language="javascript">
    t=prompt( "nhap thang: ");
    switch (eval(t))
    {
        case 1:case 3:case 5:case 7:case 8:case 10:case 12:
            alert("Thang "+ t+ " co 31 ngay"); break;
        case 2:
            alert("Thang "+t + " co 28 ngay hoặc 29 ngày");
            break;
        case 4: case 6: case 9: case 11:
            alert("Thang "+t +" co 30 ngay");break;
        default:
            alert("Khong co thang nay");break;
    }
</script>

```

Khi người dùng nhập vào một con số trong hộp thoại prompt. Nếu số nhập vào là một trong các số: 1, 3, 5, 7, 8, 10, 12 sẽ có cùng kết quả trả về là hộp thoại alert với thông báo tháng vừa nhập có 31 ngày. Nếu số mà người dùng nhập vào là số 2 thì sẽ xuất hộp thoại cho biết tháng 2 có 28 hoặc 29 ngày. Trường hợp người dùng nhập một trong các số: 4, 6, 9, 11 thì kết quả sẽ là hộp thoại cho biết tháng vừa nhập có 30 ngày. Cuối cùng là nếu người dùng

nhập các con số khác không thuộc các trường hợp trên thì đều xuất ra hộp thoại cho biết không có tháng này.

### Vòng lặp for

Với giá trị được khởi tạo ban đầu của biến, kiểm tra biến thỏa điều kiện của vòng lặp thì thực hiện công việc. Sau khi kết thúc vòng lặp, giá trị của biến sẽ được thay đổi có thể tăng hay giảm tùy theo bài toán cụ thể. Sau khi được thay đổi giá trị, biến sẽ được kiểm tra lại nếu vẫn thỏa điều kiện thì vòng lặp tiếp tục được thực hiện. Nếu không còn thỏa điều kiện thì kết thúc vòng lặp.

Cú pháp:

```
for (biểu thức 1; biểu thức 2; biểu thức 3) {  
    //Khối lệnh;  
}
```

Trong đó, biểu thức 1 thường là biểu thức khởi tạo biến, biểu thức 2 là biểu thức điều kiện, biểu thức 3 là biểu thức thay đổi giá trị của biến.

Ví dụ 3-4: vận dụng dòng lặp for vẽ lên Web một bảng biểu có m dòng, n cột được nhập từ bàn phím.

```
<html>  
<head><title> tạo bảng với m dòng, n cột nhập từ bàn phím  
</title>  
</head>  
<body>  
<script language="javascript">  
    var n, m, i, j;  
    m=prompt("Nhập số dòng");  
    n=prompt("Nhập số cột");  
    document.write("<table width=50% border=1>");  
        for(i=1;i<=m;i++)  
        {  
            document.write("<tr>");  
            for(j=1;j<=n;j++)  
                document.write("<td>" + i + j + "</td>");  
            document.write("</tr>");  
        }  
    document.write("</table>");  
</script>  
</body>  
</html>
```

Trong ví dụ này, bạn có thể thấy cách chúng tôi gọi thẻ HTML như một chuỗi trong JavaScript. Khi đoạn Script được thông dịch, trình duyệt sẽ nhận diện các thẻ HTML trong chuỗi của JavaScript và hiển thị định dạng của thẻ HTML.

#### Vòng lặp for ... in

Dòng lặp for ... in thường được dùng cho trường hợp đối tượng là mảng. Cú pháp như sau:

```
for ( variable      in      Object) {  
    // Khởi lệnh;  
}
```

#### Ví dụ 3-5: Xuất dữ liệu trong mảng bằng for ... in

```
<script language="javascript">  
    obj= new Array() ; // khai báo mảng  
    obj [0] = "Chào ";  
    obj [1] = "mừng ";  
    obj [2] = "đến ";  
    obj [3] = "với ";  
    obj [4] = "JavaScript ";  
    for(i in obj)  
        document.write(obj [i]);  
</script>
```

#### Vòng lặp while

Dòng lặp while sẽ kiểm tra điều kiện nếu thỏa thì thực hiện khối lệnh trong while cho đến khi điều kiện không còn thỏa nữa. Chú ý rằng rất có thể dòng lặp while sẽ không được thực hiện lần nào nếu biểu thức không thỏa ngay từ đầu.

#### Cú pháp:

```
while(biểu thức điều kiện){  
    //Khởi lệnh;  
}
```

#### Ví dụ 3-6: dòng lặp while

```

<script language="javascript">
    var input;
    while (input!=99)
    {
        input=prompt("Nhập vào một số bất kỳ, nhập 99 để
thoát");
        if (isNaN(input))
        {
            document.write("Dữ liệu không hợp lệ, nhập số ");
            break;
        }
    }
</script>

```

Vòng lặp do ... while

Dòng lặp do ... while sẽ thực hiện công việc trước rồi mới kiểm tra điều kiện. Có nghĩa là nếu điều kiện không thỏa thì công việc cũng đã được thực hiện ít nhất 1 lần.

Cú pháp:

```

do
{
    //khởi lệnh;
}
while (biểu thức điều kiện);

```

Ví dụ 3-7:

```

<script language="javascript">
    var input;
    do
    {
        input=prompt("Nhập vào một số bất kỳ, nhập 99 để
thoát");
        if(isNaN(input))
        {
            document.write("Dữ liệu không hợp lệ, nhập số ");
            break;
        }
    }while (input!=99)
</script>

```

### 3.2.6 Tham chiếu đến các đối tượng trong HTML

Tham chiếu đến giá trị của thẻ

Để lấy được thông tin mà người dùng đã nhập vào form để kiểm tra tính hợp lệ, chúng ta cần tham chiếu đến các thành phần của form. Để tham chiếu đến form, bạn cần chú ý đến tên form và tên của các thành phần trong form mà bạn dự định sẽ lấy dữ liệu.

THẺ	VD
Text	document.ten_form.ten_txt.value
Password	document.ten_form.ten_txtpass.value
Textarea	document.ten_form.ten_txtarea.value
Select (combobox)	document.ten_form.ten_combo.value
Select (listbox)	document.ten_form.ten_listbox.value
Checkbox	document.ten_form.ten_checkbox[i].value
Radio	document.ten_form.ten_radio[i].value
...	...

Tham chiếu đến chiều dài của thẻ

Để lấy được chiều dài của thông tin nhập liệu, chúng ta cần lấy được thông tin đó trước, rồi sau đó mới xem chiều dài của thông tin.

THẺ	VD
Text	document.ten_form.ten_txt.value.length
Password	document.ten_form.ten_txtpass.value.length
Textarea	document.ten_form.ten_txtarea.value.length
Select (combobox)	document.ten_form.ten_combo.value.length

THẺ	VD
Select (listbox)	document.ten_form.ten_listbox.value.length
Checkbox	document.ten_form.ten_checkbox[i].value.length
Radio	document.ten_form.ten_radio[i].value.length
...	...

Tham chiếu đến trạng thái của thẻ

Chỉ có một số đối tượng trong form có thể kiểm tra trạng thái.

THẺ	VD
Select (combobox)	document.ten_form.ten_combo.selected
Select (listbox)	document.ten_form.ten_listbox.selected
Checkbox	document.ten_form.ten_checkbox[i].checked
Radio	document.ten_form.ten_radio[i].checked
...	...

Nếu muốn tham chiếu đến trạng thái của một khung nhập liệu (chẳng hạn như Text Fields, Password Field, ...) có được nhập thông tin vào chưa, ta quay về với dạng tham chiếu đến giá trị của các Fields đó. Nếu sau khi tham chiếu mà trả về là rỗng tức là Field đó chưa được nhập liệu.

Một số phương thức:

PHƯƠNG THỨC	DIỄN GIẢI
document.ten_form.ten_tp_form.focus()	Đặt con trỏ nhấp nháy vào một thành phần có tên là ten_tp_form trong form có tên là ten_form

PHƯƠNG THỨC	DIỄN GIẢI
window.close()	Đóng cửa sổ hiện tại
window.open(URL)	Mở cửa sổ trình duyệt với địa chỉ URL
window.history.back(i)	Trở về cửa sổ trình duyệt trang web i lần
window.history.go(i)	Trở về cửa sổ trình duyệt trang web i lần
...	...

Thao tác trên id của thẻ HTML

Khi muốn viết nội dung vào các thành phần của form ta quay về dạng tham chiếu đến giá trị của thẻ và chỉ việc gán nội dung. Nhưng nếu đó là một thẻ HTML không thuộc thành phần của biểu mẫu, chúng ta không thể áp dụng theo cách tham chiếu giá trị được. Lúc này chúng ta sẽ sử dụng thuộc tính id để nhận dạng các thẻ HTML cần chèn thêm thông tin từ JavaScript. Sau đó sử dụng thành phần `getElementById("ten_id")`. Tuy nhiên có hai dạng truy cập vào id của một thẻ HTML.

Viết nội dung vào hoặc lấy nội dung đã trình bày sẵn trong cặp thẻ HTML đã được đặt trước một id:

```
document.getElementById("ten_ID").innerHTML
```

Thao tác với id của thẻ:

```
document.getElementById("ten_ID").id
```

Trường hợp vừa muốn thao tác lên thẻ có id được chỉ định lại vừa muốn thao tác trên nội dung bên trong của thẻ:

```
document.getElementById("ten_ID").outerHTML
```

Sử dụng một số đối tượng

Đối tượng String:

<b>Method</b>	<b>Description</b>
<u><a href="#">anchor()</a></u>	Creates an HTML anchor
<u><a href="#">big()</a></u>	Displays a string in a big font
<u><a href="#">blink()</a></u>	Displays a blinking string
<u><a href="#">bold()</a></u>	Displays a string in bold
<u><a href="#">charAt()</a></u>	Returns the character at a specified position
<u><a href="#">charCodeAt()</a></u>	Returns the Unicode of the character at a specified position
<u><a href="#">concat()</a></u>	Joins two or more strings
<u><a href="#">fixed()</a></u>	Displays a string as teletype text
<u><a href="#">fontcolor()</a></u>	Displays a string in a specified color
<u><a href="#">fontsize()</a></u>	Displays a string in a specified size
<u><a href="#">fromCharCode()</a></u>	Takes the specified Unicode values and returns a string
<u><a href="#">indexOf()</a></u>	Returns the position of the first occurrence of a specified string value in a string
<u><a href="#">italics()</a></u>	Displays a string in italic
<u><a href="#">lastIndexOf()</a></u>	Returns the position of the last occurrence of a specified string value, searching backwards from the specified position in a string
<u><a href="#">link()</a></u>	Displays a string as a hyperlink
<u><a href="#">match()</a></u>	Searches for a specified string value in a string
<u><a href="#">replace()</a></u>	Replaces some characters with some other characters in a string
<u><a href="#">search()</a></u>	Searches a string for a specified value
<u><a href="#">slice()</a></u>	Extracts a part of a string and returns the extracted part in a new string
<u><a href="#">small()</a></u>	Displays a string in a small font
<u><a href="#">split()</a></u>	Splits a string into an array of strings
<u><a href="#">strike()</a></u>	Displays a string with a strikethrough
<u><a href="#">sub()</a></u>	Displays a string as subscript
<u><a href="#">substr()</a></u>	Extracts a specified number of characters in a string, from a start index
<u><a href="#">substring()</a></u>	Extracts the characters in a string between two specified indices
<u><a href="#">sup()</a></u>	Displays a string as superscript
<u><a href="#">toLowerCase()</a></u>	Displays a string in lowercase letters
<u><a href="#">toUpperCase()</a></u>	Displays a string in uppercase letters
<u><a href="#">toSource()</a></u>	Represents the source code of an object
<u><a href="#">valueOf()</a></u>	Returns the primitive value of a String object

Ví dụ 3-8: Sử dụng một số phương thức trong đối tượng String

```
-----  
| var txt = "Hello world!";  
| document.write (txt.length);  
| document.write (txt.toUpperCase());  
| document.write (txt.substring(4,8));  
|-----
```

Đối tượng Date:

Method	Description
<a href="#">Date()</a>	Returns today's date and time
<a href="#">getDate()</a>	Returns the day of the month from a Date object (from 1-31)
<a href="#">getDay()</a>	Returns the day of the week from a Date object (from 0-6)
<a href="#">getMonth()</a>	Returns the month from a Date object (from 0-11)
<a href="#">getFullYear()</a>	Returns the year, as a four-digit number, from a Date object
<a href="#">getYear()</a>	Returns the year, as a two-digit or a four-digit number, from a Date object. Use <code>getFullYear()</code> instead !!
<a href="#">getHours()</a>	Returns the hour of a Date object (from 0-23)
<a href="#">getMinutes()</a>	Returns the minutes of a Date object (from 0-59)
<a href="#">getSeconds()</a>	Returns the seconds of a Date object (from 0-59)
<a href="#">getMilliseconds()</a>	Returns the milliseconds of a Date object (from 0-999)
<a href="#">getTime()</a>	Returns the number of milliseconds since midnight Jan 1, 1970
<a href="#">getTimezoneOffset()</a>	Returns the difference in minutes between local time and Greenwich Mean Time (GMT)
<a href="#">getUTCDate()</a>	Returns the day of the month from a Date object according to universal time (from 1-31)
<a href="#">setUTCDate()</a>	Sets the day of the month in a Date object according to universal time (from 1-31)
<a href="#">setUTCMonth()</a>	Sets the month in a Date object according to universal time (from 0-11)
<a href="#">setUTCFullYear()</a>	Sets the year in a Date object according to universal time (four digits)
<a href="#">setUTCHours()</a>	Sets the hour in a Date object according to universal time (from 0-23)
<a href="#">setUTCMinutes()</a>	Set the minutes in a Date object according to universal time (from 0-59)
<a href="#">setUTCSeconds()</a>	Set the seconds in a Date object according to universal time (from 0-59)
<a href="#">setUTCMilliseconds()</a>	Sets the milliseconds in a Date object according to universal time (from 0-999)
<a href="#">toSource()</a>	Represents the source code of an object
<a href="#">toString()</a>	Converts a Date object to a string
<a href="#">toGMTString()</a>	Converts a Date object, according to Greenwich time, to a string. Use <code>toUTCString()</code> instead !!
<a href="#">toUTCString()</a>	Converts a Date object, according to universal time, to a string
<a href="#">toLocaleString()</a>	Converts a Date object, according to local time, to a string
<a href="#">UTC()</a>	Takes a date and returns the number of milliseconds since midnight of January 1, 1970 according to universal time
<a href="#">valueOf()</a>	Returns the primitive value of a Date object

### Ví dụ 3-9: Sử dụng một số phương thức của đối tượng Date

```
-----  
| var myDate = new Date();  
| myDate.setFullYear(2010, 0, 14);  
| myDate.setDate(myDate.getDate() + 5);  
| myDate.setFullDate(2010, 0, 14);  
  
| var today = new Date();  
| if (myDate > today)  
|     alert("Hôm nay chưa đến ngày 14/1/2010");  
| else  
|     alert("Hôm nay đã qua ngày 14/1/2010");  
-----
```

Đối tượng Array:

Method	Description
<a href="#">concat()</a>	Joins two or more arrays and returns the result
<a href="#">join()</a>	Puts all the elements of an array into a string. The elements are separated by a specified delimiter
<a href="#">pop()</a>	Removes and returns the last element of an array
<a href="#">push()</a>	Adds one or more elements to the end of an array and returns the new length
<a href="#">reverse()</a>	Reverses the order of the elements in an array
<a href="#">shift()</a>	Removes and returns the first element of an array
<a href="#">slice()</a>	Returns selected elements from an existing array
<a href="#">sort()</a>	Sorts the elements of an array
<a href="#">splice()</a>	Removes and adds new elements to an array
<a href="#">toSource()</a>	Represents the source code of an object
<a href="#">toString()</a>	Converts an array to a string and returns the result
<a href="#">unshift()</a>	Adds one or more elements to the beginning of an array and returns the new length
<a href="#">valueOf()</a>	Returns the primitive value of an Array object

### Ví dụ 3-10: Sắp xếp mảng bằng phương thức sort()

```
-----  
| var arr = new Array(6)  
| arr[0] = "Jani"  
| arr[1] = "Hege"  
| arr[2] = "Stale"  
| arr[3] = "Kai Jim"  
| arr[4] = "Borge"  
| arr[5] = "Tove"  
| document.write(arr + "<br />")  
| document.write(arr.sort())  
-----
```

Ví dụ 3-11: Sử dụng phương thức ghép chuỗi

```
-----  
var arr = new Array(3)  
arr[0] = "Jani"  
arr[1] = "Tove"  
arr[2] = "Hege"  
var arr2 = new Array(3)  
arr2[0] = "John"  
arr2[1] = "Andy"  
arr2[2] = "Wendy"  
document.write(arr.concat(arr2))  
-----
```

Đối tượng Math:

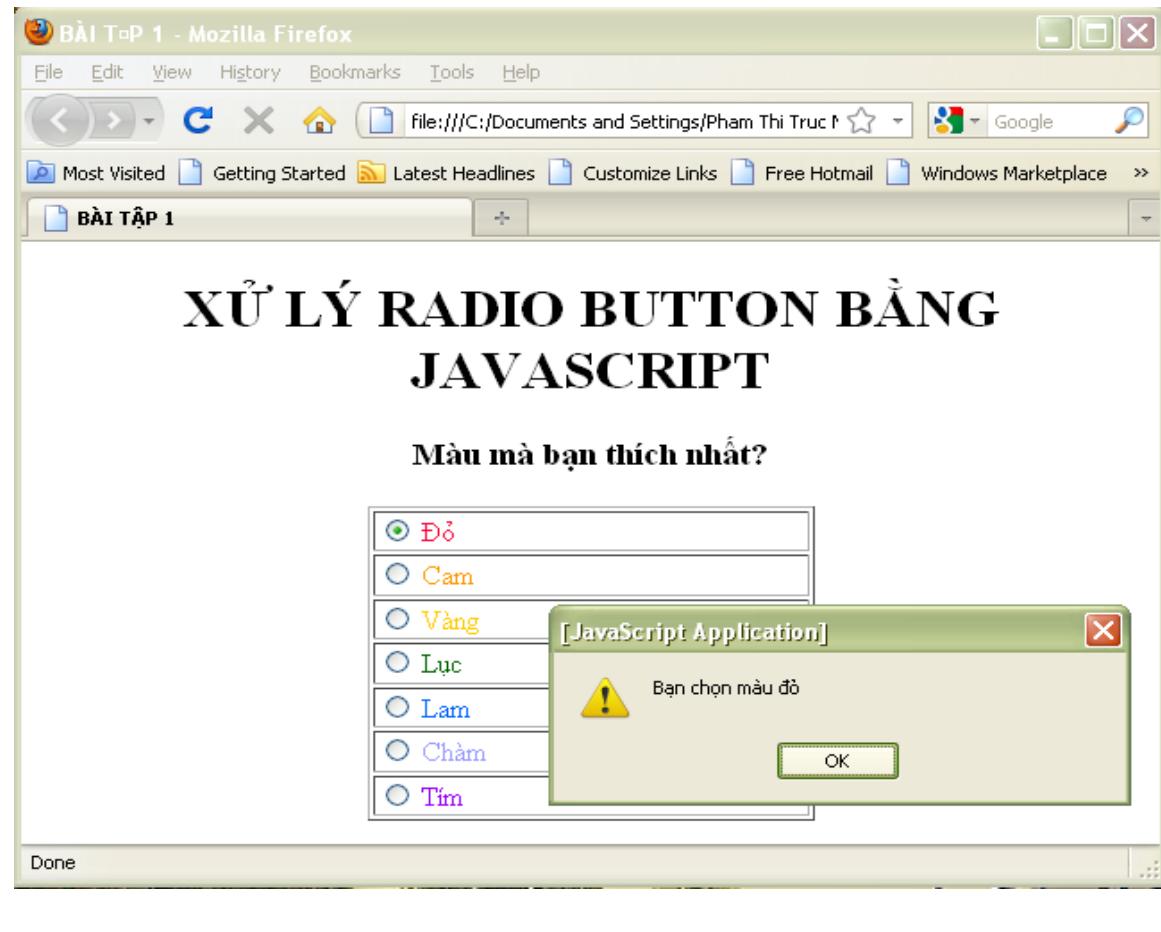
<b>Method</b>	<b>Description</b>
<a href="#"><u>abs(x)</u></a>	Returns the absolute value of a number
<a href="#"><u>acos(x)</u></a>	Returns the arccosine of a number
<a href="#"><u>asin(x)</u></a>	Returns the arcsine of a number
<a href="#"><u>atan(x)</u></a>	Returns the arctangent of x as a numeric value between -PI/2 and PI/2 radians
<a href="#"><u>atan2(y,x)</u></a>	Returns the angle theta of an (x,y) point as a numeric value between -PI and PI radians
<a href="#"><u>ceil(x)</u></a>	Returns the value of a number rounded upwards to the nearest integer
<a href="#"><u>cos(x)</u></a>	Returns the cosine of a number
<a href="#"><u>exp(x)</u></a>	Returns the value of E <sup>x</sup>
<a href="#"><u>floor(x)</u></a>	Returns the value of a number rounded downwards to the nearest integer
<a href="#"><u>log(x)</u></a>	Returns the natural logarithm (base E) of a number
<a href="#"><u>max(x,y)</u></a>	Returns the number with the highest value of x and y
<a href="#"><u>min(x,y)</u></a>	Returns the number with the lowest value of x and y
<a href="#"><u>pow(x,y)</u></a>	Returns the value of x to the power of y
<a href="#"><u>random()</u></a>	Returns a random number between 0 and 1
<a href="#"><u>round(x)</u></a>	Rounds a number to the nearest integer
<a href="#"><u>sin(x)</u></a>	Returns the sine of a number
<a href="#"><u>sqrt(x)</u></a>	Returns the square root of a number
<a href="#"><u>tan(x)</u></a>	Returns the tangent of an angle
<a href="#"><u>toSource()</u></a>	Represents the source code of an object
<a href="#"><u>valueOf()</u></a>	Returns the primitive value of a Math object

Ví dụ 3-12:

```
Math.E  
Math.PI  
Math.SQRT2  
Math.SQRT1_2  
Math.LN2  
Math.LN10  
Math.LOG2E  
Math.LOG10E  
document.write(Math.round(4.7))  
document.write(Math.random())  
document.write(Math.floor(Math.random()*11))  
document.write(Math.round(-4.60))
```

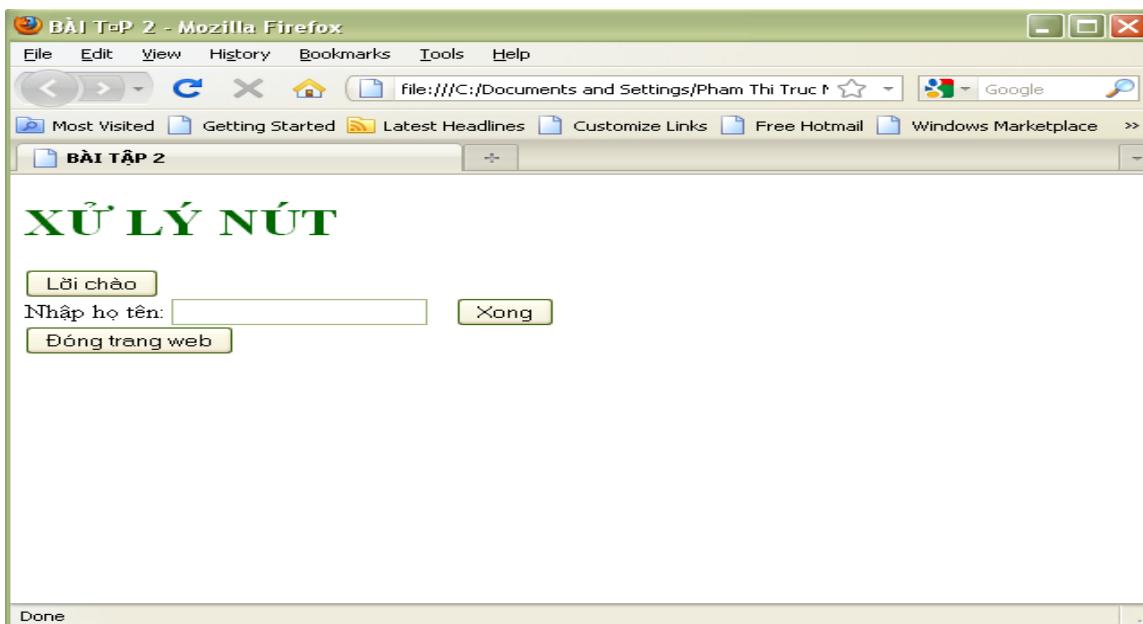
❖ Câu hỏi (bài tập) củng cố:

1. Hãy thiết kế trang HTML như sau. Yêu cầu: Khi click vào mục chọn màu nào thì hiện lên thông báo rằng bạn đã chọn màu đó. Lưu bài tập với tên JS\_baitap1.html



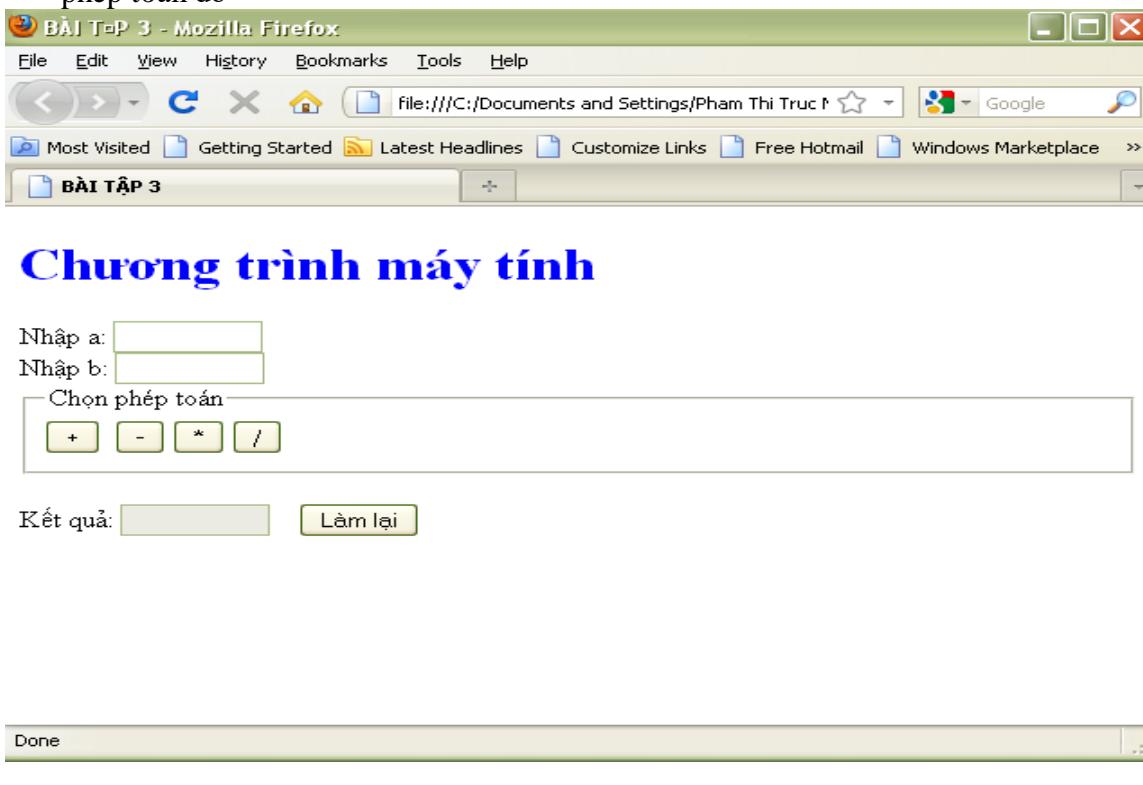
2. Hãy thiết kế trang HTML như sau, lưu bài tập với tên JS\_baitap2.html. Dùng Javascript để xử lý các nút:

- Nút lời chào, khi click vào hiện ra lời chào: Chào mừng bạn đến với Javascript.
- Khi bạn nhập họ tên vào textbox, click vào nút Xong, hộp thoại hiện ra tên bạn vừa nhập.
- Khi click vào nút đóng trang web, website của bạn sẽ bị đóng lại.



3. Viết chương trình mô phỏng máy tính điện tử. Lưu bài với tên JS\_baitap3.html.

- Sau khi bạn nhập vào 2 số a, b. Bạn hãy chọn một phép toán và xem kết quả của phép toán đó

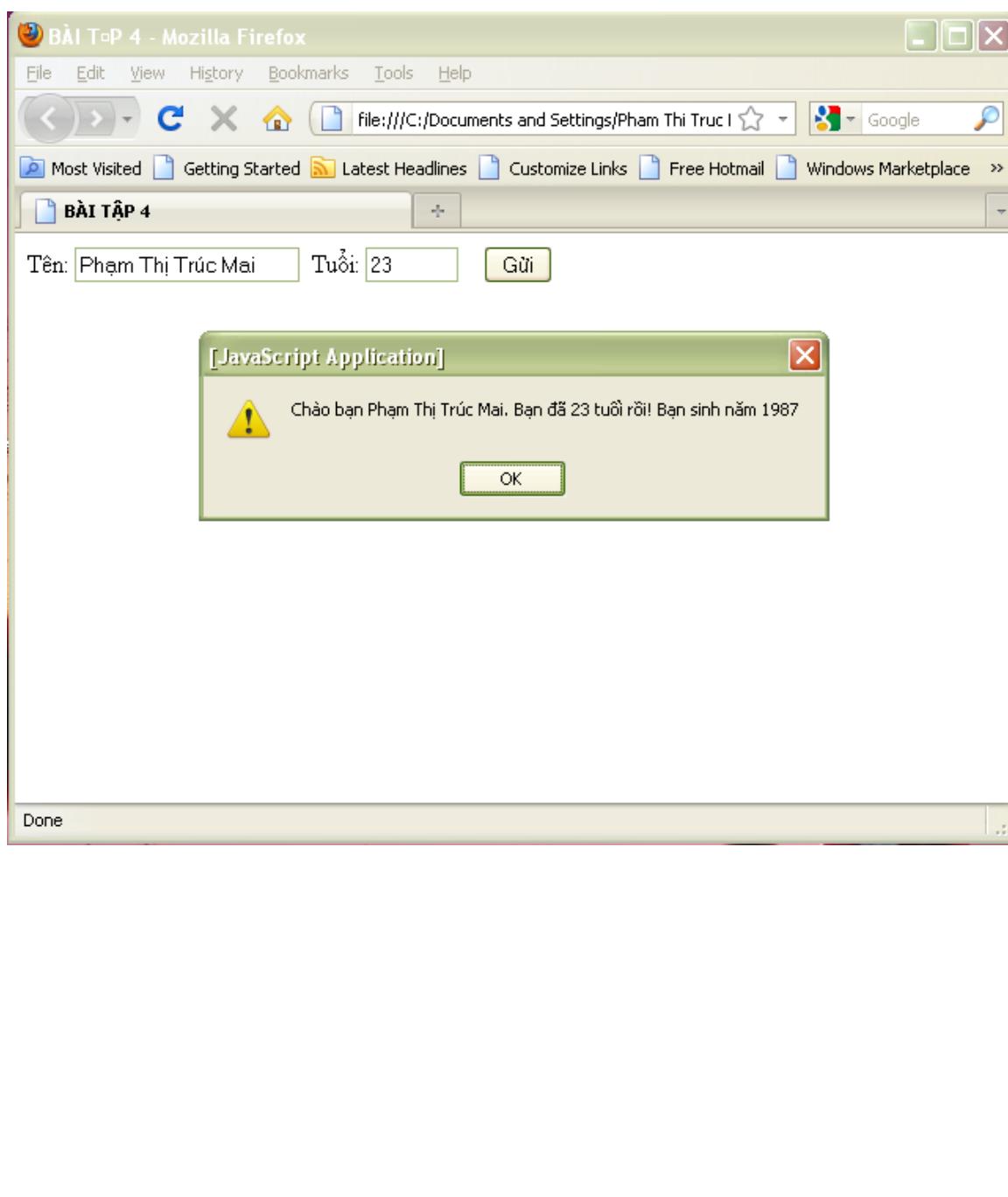


4. Bạn hãy thiết kế form như sau, lưu bài tập với tên JS\_baitap4.html

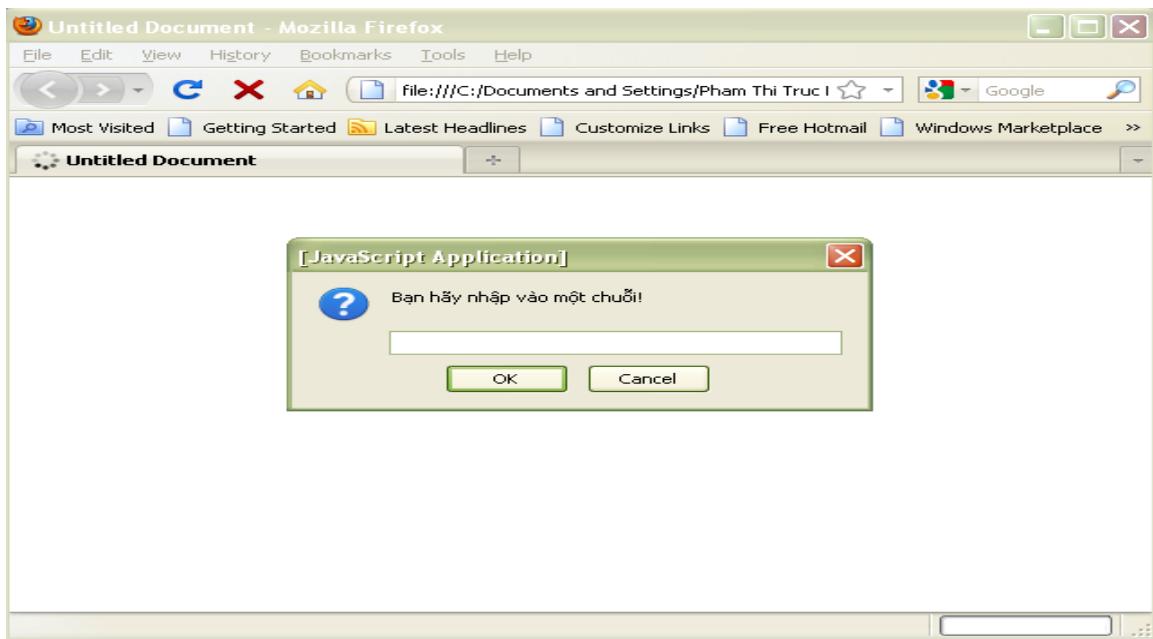
Lưu ý:

- Nếu người dùng chưa nhập tên, chương trình xuất hiện thông báo. Sau đó con trỏ chuột xuất hiện trong ô cần nhập tên.
- Tương tự, hãy lập trình cho ô nhập tuổi.
- Đặc biệt, nếu người dùng nhập tuổi là ký tự hoặc số âm thì hiện thông báo lỗi, yêu cầu nhập lại, xóa bỏ dữ liệu trong ô tuổi cũ, đặt con trỏ chuột vào ô tuổi để người dùng nhập lại.

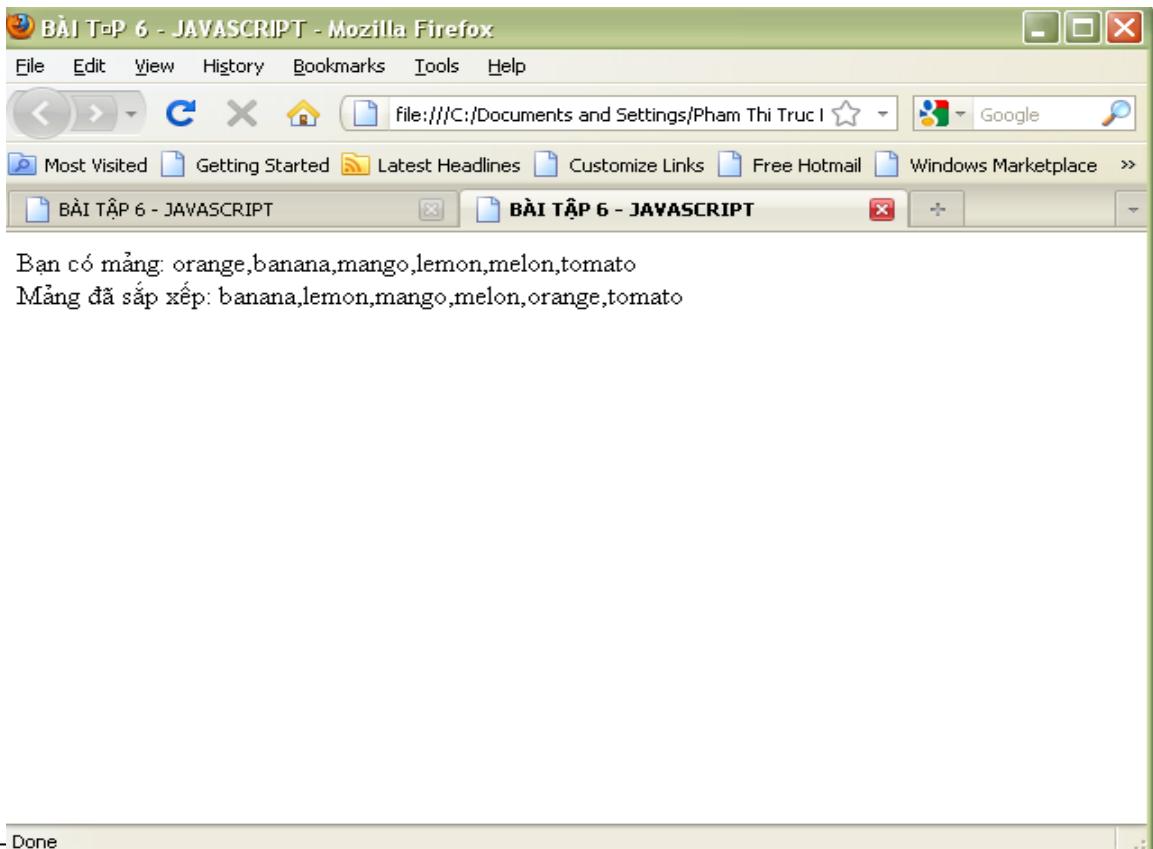
Nếu người dùng nhập đúng thông tin, click vào nút gửi, sẽ hiện lên thông tin về tên, tuổi, năm sinh của người dùng.



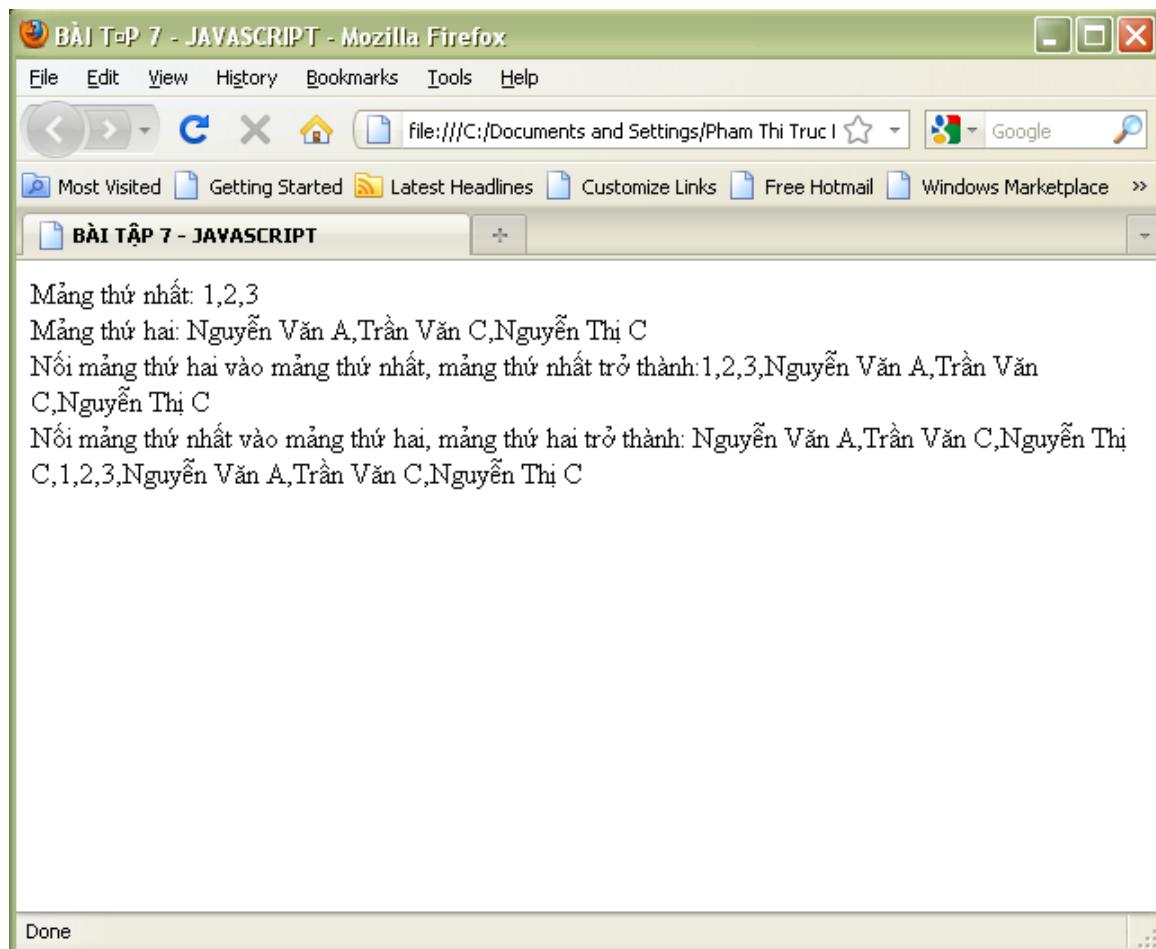
5. Hãy viết một thông báo yêu cầu người dùng nhập vào một chuỗi. Sau đó kiểm tra xem chuỗi vừa nhập có bao nhiêu ký tự. Lưu bài tập với tên JS\_baitap5.html



6. Cho trước một mảng gồm danh sách các trái cây: orange, banana, mango, lemon, melon, tomato. Hãy sắp xếp lại theo thứ tự abc. Lưu bài tập với tên JS\_baitap6.html.



7. Cho hai mảng, mảng thứ nhất gồm các phần tử: 1, 2, 3. Mảng thứ hai gồm các phần tử: Nguyễn Văn A, Trần Văn C, Nguyễn Thị C  
Thực hiện nối hai chuỗi đó lại. Lưu bài tập với tên JS\_baitap7.html.



8. Hãy tạo trang JS\_baitap8.html.

Chú ý:

- Kiểm tra tên đăng nhập không thể rỗng và nếu chiều dài khác 9 ký tự thì báo lỗi
- Kiểm tra mật khẩu phải có ít nhất 6 ký tự và nhiều nhất là 12 ký tự.
- Sau khi điền thông tin và click vào nút nhập, thông tin vừa nhập sẽ được điền vào các cột tương ứng của table.
- Hãy trang trí lại trang theo sở thích của bạn.

THÔNG TIN SINH VIÊN - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address C:\Documents and Settings\Pham Thi Truc Mai\Desktop\BAIGIANG\_TKW\_TRUCMAI\TH\_TKWEB\

**THÔNG TIN ĐĂNG KÝ**

Họ và tên:

Tên đăng nhập:

Mật khẩu:

Quê quán:  Hà Nội

**Thông tin bạn vừa nhập**

Họ tên	Tên đăng nhập	Mật khẩu	Quê quán

Done My Computer

9. Hãy tạo trang JS\_baitap9.html.

Yêu cầu:

- Sau mỗi lần khách hàng nhập vào số lượng bức tranh thì tổng thành tiền được cập nhật.

$$\text{- Tổng thành tiền} = \sum(\text{đơn giá} * \text{số lượng})$$

**TRANH CHỢ - Microsoft Internet Explorer**

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Favorites Back Forward Stop Home Search Favorites Favorites

Address C:\Documents and Settings\Pham Thi Truc Mai\Desktop\BAIGIANG\_TKW\_TRUCMAI\TH\_TKWEB\

Shop tranh chữ thư pháp

STT	Hình ảnh	Giá	Số lượng mua
1		90000	<input type="text"/>
2		200000	<input type="text"/>
3		150000	<input type="text"/>
4		120000	<input type="text"/>
5		100000	<input type="text"/>
Tổng thành tiền:			<input type="text"/>

Done My Computer

## CHƯƠNG 4

### TỔNG QUAN VỀ PHP

#### BÀI 1

#### **GIỚI THIỆU VỀ PHP**

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Nắm được khái niệm về ngôn ngữ lập trình PHP
- Giới thiệu cơ chế hoạt động của web tĩnh, web động

#### **4.1.1. PHP là gì?**

PHP có tên gọi đầy đủ là **Personal Home Page** hay **Hypertext Preprocessor** là ngôn ngữ kịch bản trên Server (Server Side Script), được thực thi trên WebServer và kết quả được gửi trả về trình duyệt Web theo yêu cầu của người dùng.

PHP là một ngôn ngữ lập trình web động mã nguồn mở. Tập tin PHP có phần mở rộng là .php. Về tổng quan PHP có cú pháp khá tương đồng với một số ngôn ngữ như C, java. Tuy nhiên, tự bản thân chúng cũng có những điểm rất riêng biệt.

#### **4.1.2. Lịch sử phát triển PHP**

PHP ra đời vào khoảng năm 1994 do Rasmus Lerdorf tạo ra với tên gọi là PHP/FI (Personal Home Page/Forms Interpreter) để theo dõi người dùng truy cập lý lịch trực tuyến của ông, được phát triển để phát sinh các form đăng nhập sử dụng giao thức HTTP của Unix.

PHP/FI 2 được công bố vào tháng 11 năm 1997 chuyển sang ngôn ngữ Script xử lý trên Server hỗ trợ cơ sở dữ liệu, Upload File, khai báo biến, mảng, câu lệnh điều kiện, biểu thức...

Nhưng không lâu sau đó, nhận ra được sự yếu kém của PHP/FI 2.0 trong việc phát triển các ứng dụng thương mại điện tử, Andi Gutmans và Zeev Suraski đã viết lại toàn bộ mã nguồn. PHP 3.0 đã chính thức được công bố vào tháng 6 năm 1998, sau thời gian 9 tháng được cộng đồng kiểm nghiệm, được xem là thế hệ kế tiếp của PHP/FI 2.0 và chấm dứt phát triển PHP/FI 2.0. Một trong những sức mạnh lớn nhất của PHP 3.0 là các tính năng mở rộng mạnh mẽ của nó. Ngoài khả năng cung cấp cho người dùng cuối một cơ sở hạ tầng chặt chẽ

dùng cho nhiều cơ sở dữ liệu, giao thức, các giao diện lập trình ứng dụng (API - Application Programming Interface); các tính năng mở rộng của PHP 3.0 đã thu hút rất nhiều nhà phát triển tham gia và đề xuất các mô-đun mở rộng mới. Hoàn toàn có thể kết luận được rằng đây chính là điểm mấu chốt dẫn đến thành công vang dội của PHP 3.0. Các tính năng khác được giới thiệu trong PHP 3.0 gồm có hỗ trợ cú pháp hướng đối tượng và nhiều cú pháp ngôn ngữ nhất quán khác, làm việc được trên nhiều hệ điều hành, giao thức email (SNMP, IMAP), ... Ngôn ngữ hoàn toàn mới đã được công bố dưới một cái tên mới, xóa bỏ mối liên hệ với việc sử dụng vào mục đích cá nhân hạn hẹp mà cái tên PHP/FI 2.0 gợi nhắc. Nó đã được đặt tên ngắn gọn là 'PHP', một kiểu viết tắt hồi quy của "PHP: Hypertext Preprocessor". Vào thời kì đỉnh cao, PHP 3.0 đã được cài đặt cho xấp xỉ 10% số máy chủ Web có trên mạng Internet.

Tháng 5 năm 2000, PHP 4 ra đời nhằm cải tiến tốc độ xử lý các ứng dụng phức tạp và cải tiến tính mô đun của cơ sở mã PHP. Những ứng dụng như vậy đã chạy được trên PHP 3.0 dựa trên các tính năng mới và sự hỗ trợ khá nhiều các cơ sở dữ liệu và API của bên thứ ba, nhưng PHP 3.0 đã không được thiết kế để xử lý các ứng dụng phức tạp như thế này một cách có hiệu quả. Ngoài tốc độ được cải thiện rất nhiều, PHP 4 đem đến các tính năng chủ yếu khác gồm có sự hỗ trợ nhiều máy chủ Web hơn, hỗ trợ phiên làm việc HTTP, tạo bộ đệm thông tin đầu ra, nhiều cách xử lý thông tin người sử dụng nhập vào bảo mật. Với PHP 4, số nhà phát triển dùng PHP đã lên đến hàng trăm nghìn và hàng triệu site đã công bố cài đặt PHP, chiếm khoảng 20% số tên miền trên mạng Internet.

Năm 2005 (PHP 5): Bổ sung Zend Engine II hỗ trợ lập trình hướng đối tượng, XML, SOAP cho Web Services, SQLite.

Hiện nay phiên bản tiếp theo của PHP đang được phát triển và hi vọng sẽ lắp đầy những khuyết điểm của phiên bản hiện tại. Ví dụ: Hỗ trợ namespace, hỗ trợ Unicode...

### **4.1.3. Ưu điểm và nhược điểm của PHP**

#### **4.1.3.1. Ưu điểm**

Là một ngôn ngữ mã nguồn mở, PHP được một số lớn các nhà lập trình Web chọn làm ngôn ngữ để phát triển Website nhờ vào một số ưu điểm nổi bật sau:

Đầu tiên, điều dễ nhận thấy là PHP có cấu trúc lệnh quen thuộc, dễ học vì nó có cú pháp gần giống ngôn ngữ C – một ngôn ngữ lập trình mà hầu hết các nhà lập trình đều biết đến.

Thứ hai, như đã giới thiệu PHP là một ngôn ngữ viết web mã nguồn mở và nó hoàn toàn miễn phí. Đó là lý do tại sao PHP lại được sử dụng rộng rãi trong môi trường phát triển web hiện nay. Bạn có thể tải nó trên trang <http://www.php.net> và cài đặt vào máy tính của mình bằng một vài bước khá đơn giản. Bên cạnh đó ta lại có rất nhiều phần mềm hỗ trợ soạn thảo web khá mạnh. Nếu như trước đây, chúng ta phải lập trình trên trình soạn thảo văn bản (Notepad) thì ngày nay đã có rất nhiều phần mềm hỗ trợ soạn thảo web bằng ngôn ngữ PHP hoàn toàn miễn phí như PHP Coder, jEdit, Notepad ++, ... hoặc các phần mềm chuyên dụng để thiết kế web như Macromedia Dreamweaver, Microsoft FrontPage, ...

Ưu điểm thứ ba là PHP được hỗ trợ tốt trên các WebServer phổ biến hiện nay như Apache, Netscape, ...

Thứ tư là PHP có thể kết hợp với nhiều hệ quản trị cơ sở dữ liệu khác nhau như: MySQL, PostgreSQL, Informix, IBM DB2, Oracle, Sybase, SQLite, Unix dbm,... điều này rất có lợi trong việc lựa chọn hệ quản trị cơ sở dữ liệu phù hợp với điều kiện cụ thể của mỗi nhà lập trình web.

Thứ năm là PHP là một ngôn ngữ đa nền, đa môi trường (Multi-Platform). Nó có thể chạy được trên nhiều hệ điều hành khác nhau: Unix, Mac OSX, Windows NT/98/2000/XP/2003/Vista/7

Cuối cùng là cùng với sự phát triển của các phần mềm tích hợp sẵn bộ ba: Apache, PHP, MySQL nhỏ gọn, thông dụng và tiện lợi. Quá trình cài đặt WebServer, hệ quản trị cơ sở dữ liệu MySQL và PHP trở nên vô cùng đơn giản giúp các nhà lập trình Web có thể rút ngắn thời gian cài đặt và cấu hình trên máy cục bộ. Một số phần mềm WebServer được tích hợp sẵn thường dùng như: WampServer, EasyPHP, Xamp, AppServer...

#### **4.1.3.2. Nhược điểm:**

Bên cạnh những ưu điểm đã trình bày ở trên, PHP cũng có nhược điểm không thể chối bỏ. Đó là PHP được tạo ra chỉ để phục vụ một mục đích duy nhất là phát triển ứng dụng web. Do đó, bạn không thể dùng ngôn ngữ này để phát triển phần mềm ứng dụng. Nhược điểm này làm cho PHP mất đi một số lượng khá lớn khách hàng khi họ chỉ muốn phát triển ứng dụng chạy trên window-form.

#### **4.1.4. Cơ chế hoạt động của Web động**

Trong Chương 1, chúng ta đã được giới thiệu cơ bản về cách hoạt động của một trang

Web. Tuy nhiên, trong thực tế, Web được xây dựng theo hai hướng: tĩnh và động.

Một trang web chỉ có các thẻ HTML, JavaScript chỉ là một trang web tĩnh. Chúng rất ít có sự tương tác với người dùng. Nếu có cũng chỉ là các bước xử lý ở mức Client, ngay tại chính trình duyệt của người dùng như kiểm tra dữ liệu hoặc chỉ hiển thị cục bộ những gì trình duyệt xử lý được từ thông tin mà người dùng nhập vào. Điều này rất hạn chế khi các thông tin của bạn chỉ có thể ở phía cục bộ, không thỏa mãn nhu cầu chia sẻ thông tin. Hơn thế nữa, những gì bạn muốn hiển thị trên web tĩnh sẽ được viết trực tiếp trong trang HTML. Chính vì thế thông tin hiển thị cũng dễ bị lỗi thời, khó cập nhật và tốn nhiều thời gian trong quá trình soạn thảo cũng như cập nhật mới.

Một hướng khác để xây website đó là có sự tương tác thật sự giữa người dùng và trang web. Người dùng có thể đăng ký làm thành viên của một Website, có thể đăng nhập hệ thống website, thêm, sửa, xóa một số thông tin mà mình được cấp quyền. Những người dùng khác có thể xem những thay đổi đó trên website (nếu được phép). Người ta gọi đây là các trang web động. Như vậy, đòi hỏi web động phải có một cơ sở dữ liệu đi kèm để lưu trữ những hoạt động trên hệ thống Website. Như vậy, điều dễ nhận thấy sự khác biệt giữa web tĩnh và web động đó chính là về mặt lưu trữ và hiển thị thông tin lưu trữ.

Bên cạnh đó, web tĩnh và web động còn khác nhau về công cụ xây dựng. Như đã trình bày, web tĩnh được viết bằng ngôn ngữ HTML, CSS, JavaScript, ... Web động bên cạnh chấp nhận tất cả các ngôn ngữ xây dựng web tĩnh, nó còn có ngôn ngữ dùng để xử lý dữ liệu chạy trên máy chủ: PHP, JSP, ASP, ...

#### ❖ Câu hỏi (bài tập) củng cố:

1. Từ viết tắt của PHP là gì?
2. Sinh viên hãy cho biết những ưu điểm và nhược điểm của PHP.
3. Để học tốt PHP, sinh viên cần chuẩn bị những kiến thức gì?
4. Sinh viên hãy cho biết sự giống nhau và khác nhau giữa web động và web tĩnh.

## BÀI 2

### **GIỚI THIỆU VÀ CÀI ĐẶT WEB SERVER**

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Hiểu được khái niệm webserver
- Cài đặt cấu hình webserver

#### **4.2.1. Giới thiệu WebServer**

WebServer hay còn gọi là trình phục vụ Web là một máy tính đóng vai trò làm máy chủ có dung lượng lớn, tốc độ xử lý cao và có khả năng lưu trữ thông tin như một ngân hàng dữ liệu, chứa những Website đã được thiết kế cùng với những thông tin liên quan khác: các mã kịch bản, các chương trình, và các tập tin đa phương tiện, ... Khi nhận được yêu cầu từ một trình duyệt Web trên máy người dùng, WebServer sẽ xử lý yêu cầu đó và có khả năng gửi đến máy khách những trang Web thông qua môi trường Internet (hoặc Intranet) qua giao thức HTTP - giao thức được thiết kế để gửi các tập tin đến trình duyệt Web (Web Browser), và các giao thức khác.

Tất cả các WebServer đều có một địa chỉ IP hay một tên miền. Giả sử khi bạn gõ vào thanh địa chỉ của trình duyệt: <http://www.tvu.edu.vn> và nhấn Enter. Yêu cầu của bạn sẽ được chuyển đến Server đang chứa Website của Trường Đại học Trà Vinh dựa vào tên miền mà bạn đã gõ. Server này sẽ tìm trang Web có tên là index, nếu bên trong tập tin này có kèm theo các mã HTML và các Client Script ví dụ như JavaScript, thì Server sẽ gửi nó đến trình duyệt của bạn. Trình duyệt của bạn sẽ làm nhiệm vụ dịch và hiển thị nội dung, định dạng của trang theo ý định của người lập trình. Nếu trong tập tin bạn yêu cầu có các Server Script như PHP, JSP, ASP, ... thì trình phục vụ Web sẽ dịch và gửi kết quả về cho trình duyệt của bạn hiển thị.

Bất kỳ một máy tính nào cũng có thể trở thành một WebServer bởi việc cài đặt lên nó một chương trình phần mềm WebServer Software: IIS, Apache, Netscape, ... và sau đó kết nối vào Internet là không quá khó. Tuy nhiên không phải máy tính nào cũng được chọn làm WebServer vì người ta còn quan tâm đến khả năng phục vụ, tốc độ, ... Giống như những phần mềm khác mà bạn đã từng cài đặt trên máy tính của mình, WebServer Software cũng chỉ là một phần mềm ứng dụng. Nó được cài đặt, và chạy trên máy tính dùng làm WebServer,

nhờ có chương trình này mà người sử dụng có thể truy cập đến các trang Web từ một máy tính khác ở trên mạng. Ngày nay, WebServer Software còn có thể được tích hợp với cơ sở dữ liệu (Database), hay điều khiển việc kết nối vào cơ sở dữ liệu được chứa trên một Database Server khác. Một số WebServer Software được tích hợp sẵn, có thể chạy trên máy tính cục bộ: WampServer, EasyPHP, AppServ, Xamp, ...

Trong khuôn khổ tài liệu này, chúng tôi sẽ sử dụng bộ ba: PHP, Apache và MySQL để thực hiện xây dựng web động. Sau đây chúng tôi sẽ hướng dẫn các bạn cài đặt WampServer. Đây là một sản phẩm tích hợp bộ ba nói trên và chạy tốt trên hệ điều hành Windows. Bạn có thể tải phiên bản mới nhất của WampServer tại địa chỉ: <http://www.wampserver.com/en/> và lựa chọn phiên bản tương ứng với hệ điều hành đang sử dụng.

<b>DOWNLOAD WampServer 2.1e (32 bits)</b>  (december 27 2010)	<b>DOWNLOAD WampServer 2.1d (64 bits)</b>  (december 27 2010)
Apache 2.2.17	Apache 2.2.17
Php 5.3.5	Php 5.3.4
Mysql 5.5.8	Mysql 5.1.53
PhpMyadmin 3.2.0.1	PhpMyadmin 3.2.0.1
SQLBuddy 1.3.2	SQLBuddy 1.3.2
XDebug 2.1.0-5.3	XDebug 2.1.0-5.3
webGrind 1.0	webGrind 1.0
XDC 1.5	XDC 1.5
<b>taille: 36Mo</b>	<b>taille: 36Mo</b>

**Hình 4-2-1:Lựa chọn phiên bản WampServer để tải về**

Tập tin tải về có tên là **WampServer2.1d-x64.exe** với dung lượng **25Mb**.

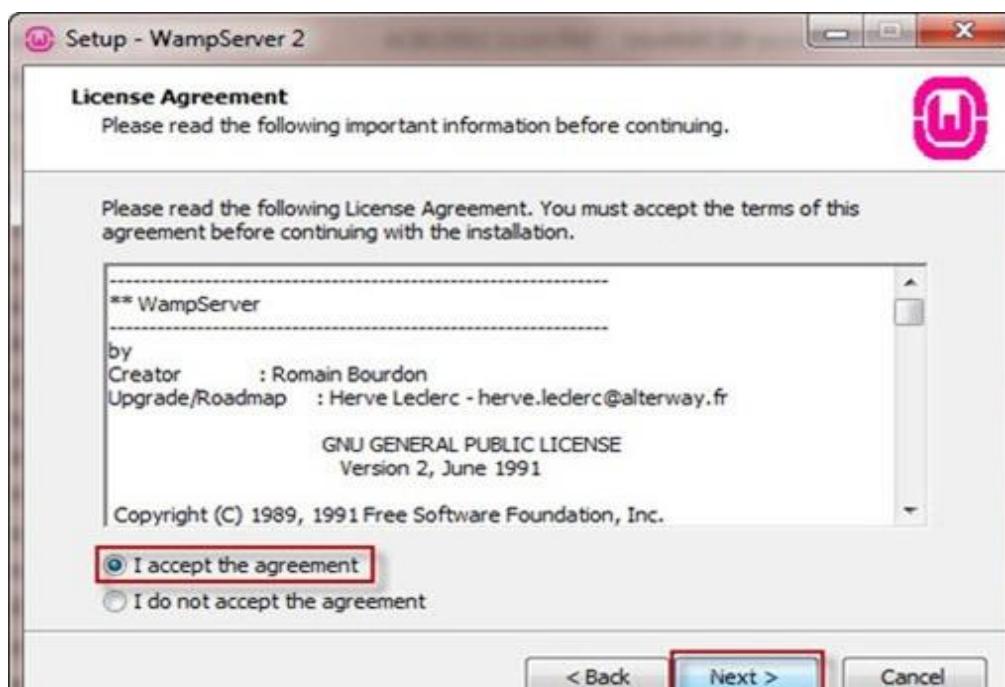
#### **4.2.2. Cài đặt WampServer2**

Để bắt đầu cài đặt WampServer 2 bạn nhấp đúp vào tập tin WampServer2.1d-x64.exe vừa tải về, sau đó chọn **Next** để bắt đầu quá trình cài đặt.



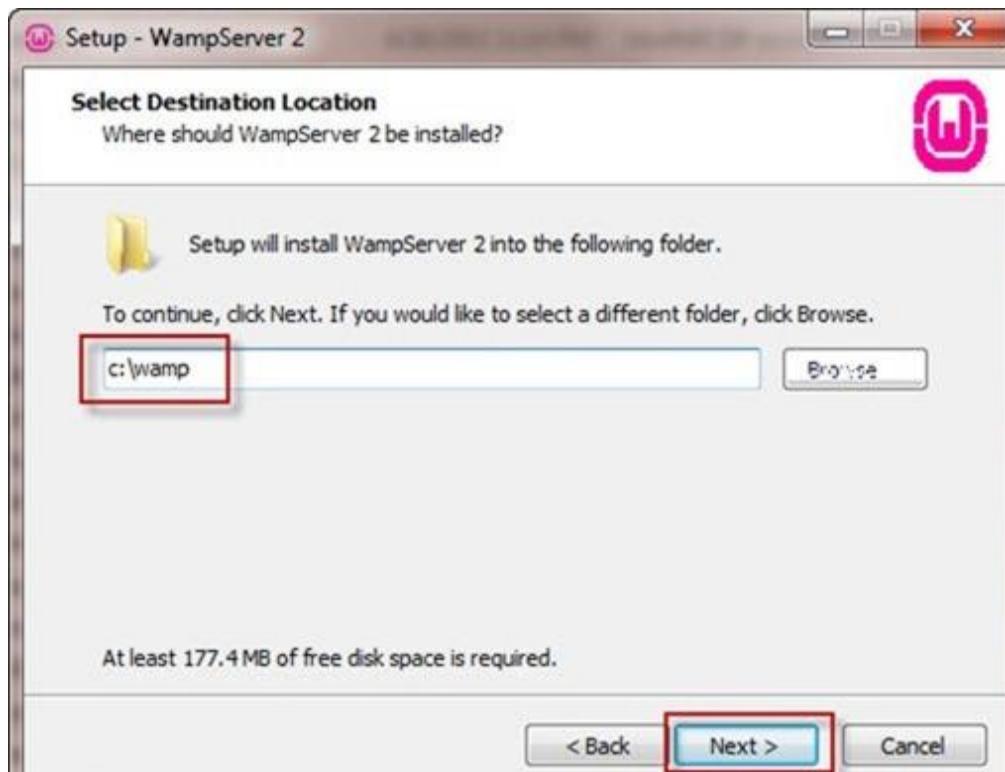
Hình 4-2-2: Bước 1 – quá trình cài đặt WampServer2

Bạn chọn **I accept the agreement** và nhấn **Next** để tiếp tục:



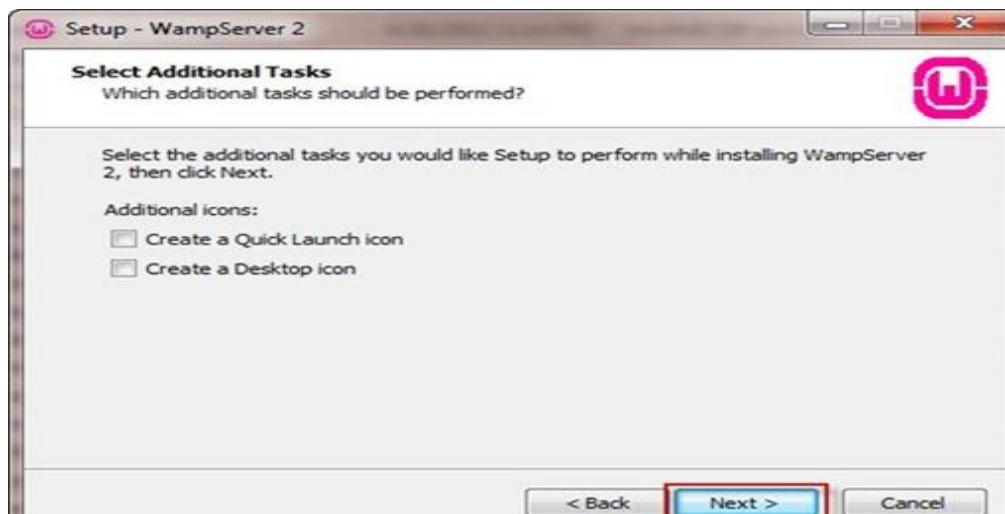
Hình 4-2-3: Bước 2 - quá trình cài đặt WampServer2

Thư mục cài đặt mặc định của WampServer 2 là **C:\wamp**, nếu muốn thay đổi bạn nhấn vào nút **Browse** và trả tới thư mục mới sau đó nhấn **Next** để tiếp tục:



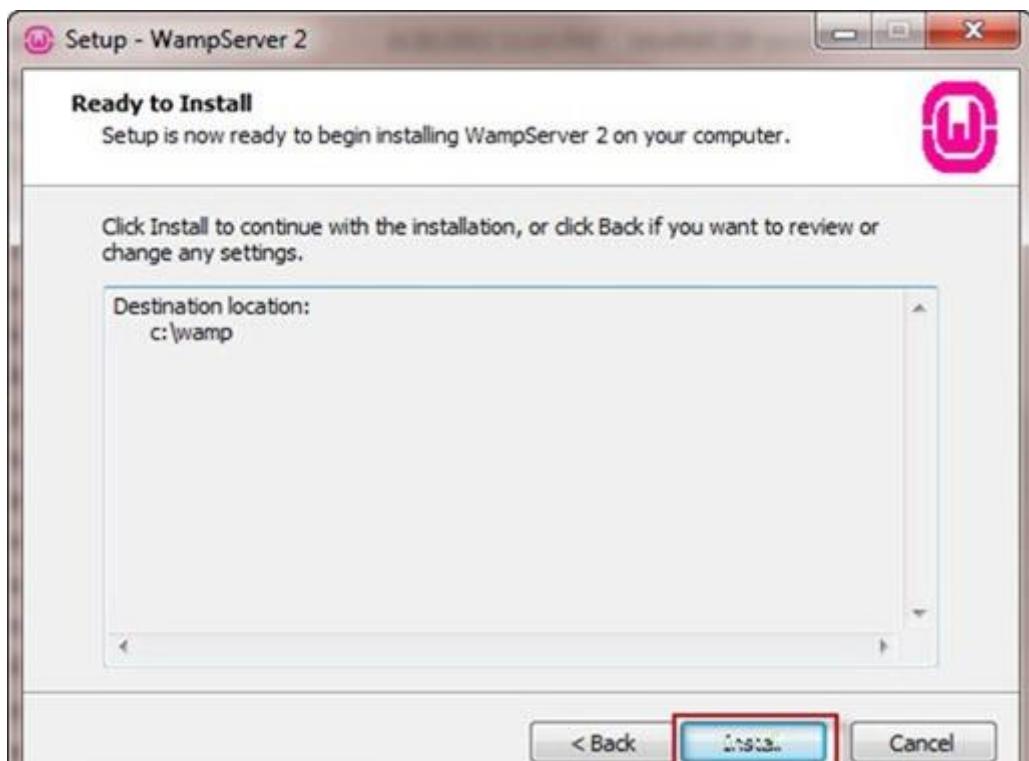
**Hình 4-2-4: Bước 3 - quá trình cài đặt WampServer2**

Chọn **Next** ở bước tiếp theo:



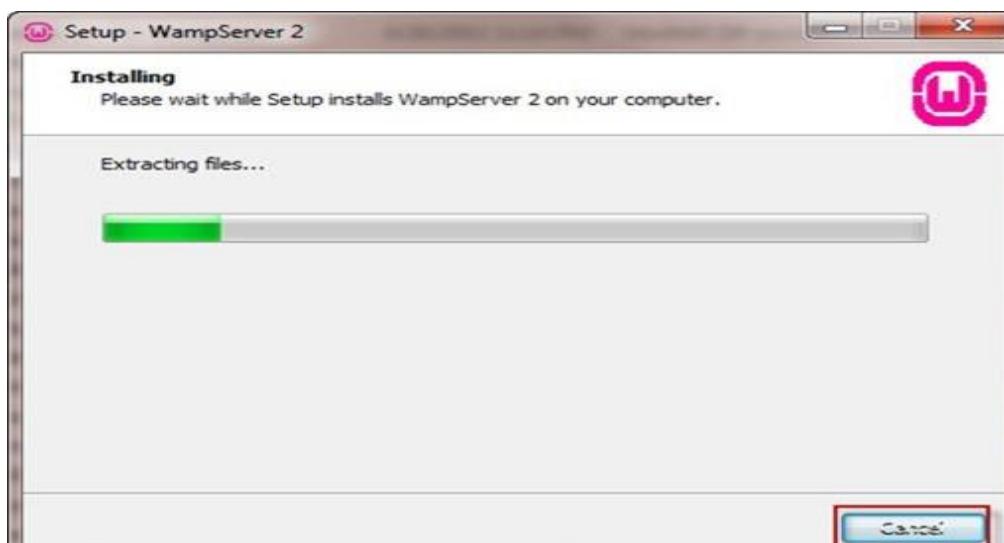
**Hình 4-2-5: Bước 4 - quá trình cài đặt WampServer2**

Chọn **Install** để bắt đầu quá trình cài đặt WampServer 2



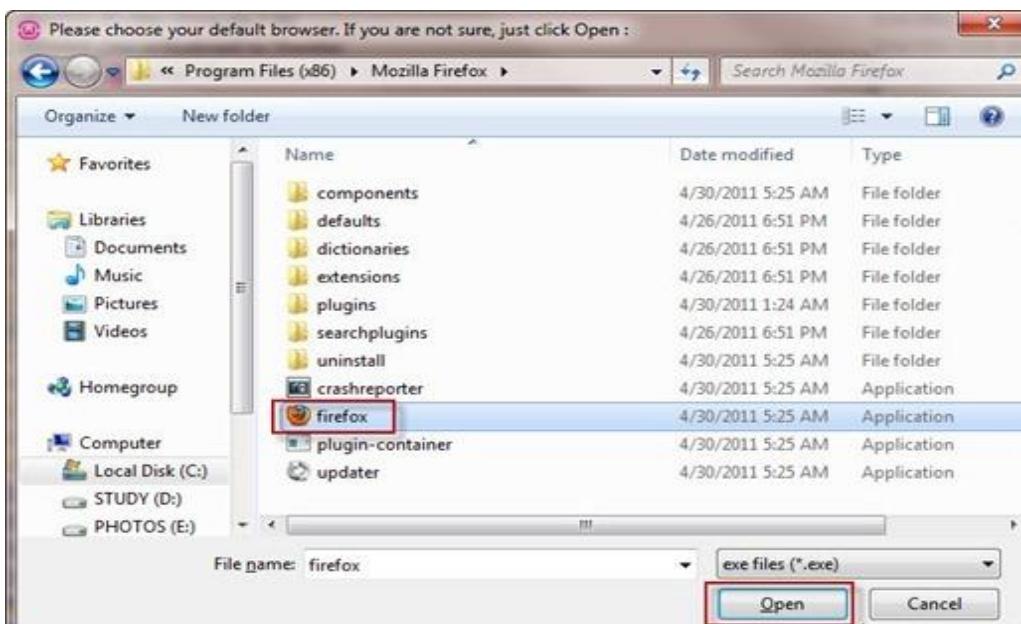
**Hình 4-2-6: Bước 5 - quá trình cài đặt WampServer2**

Màn hình tiếp theo thông báo tới bạn tiến độ cài đặt WampServer 2



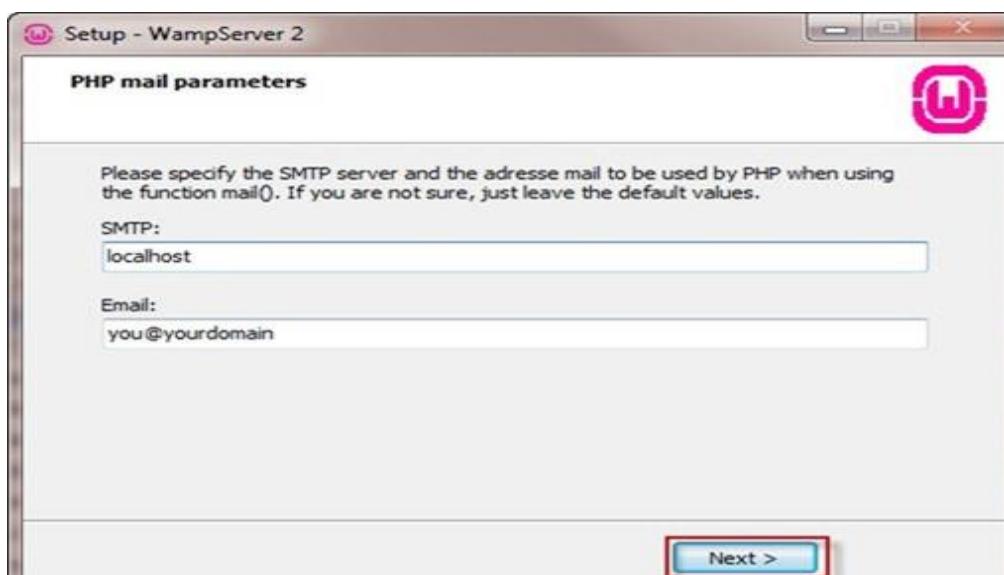
**Hình 4-2-7: Bước 6 - quá trình cài đặt WampServer2**

Trong quá trình cài đặt sẽ có bước WampServer 2 yêu cầu bạn chọn trình duyệt web mặc định cho ứng dụng, bạn nhấn **Open** và trả lời nơi cài đặt trình duyệt web của bạn ở mục **Program Files**.



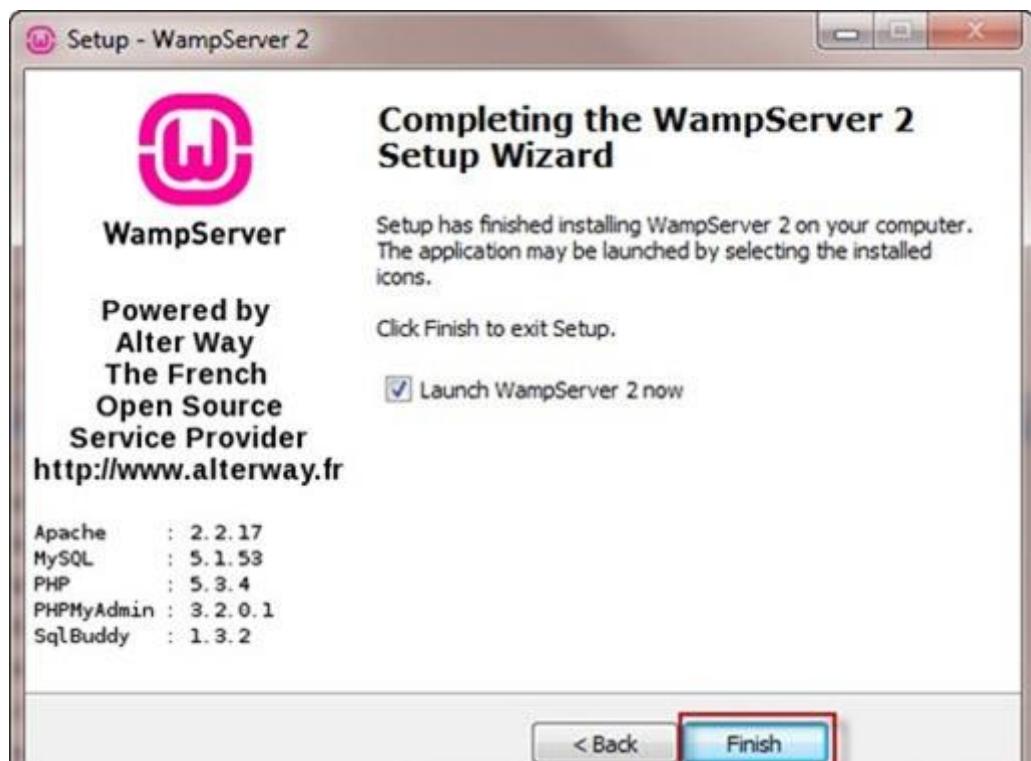
Hình 4-2-8: Bước 7 - quá trình cài đặt WampServer2

Chọn **Next** để tiếp tục



Hình 4-2-9: Bước 8 - quá trình cài đặt WampServer2

Quá trình cài đặt hoàn tất bạn chọn **Finish** để kết thúc

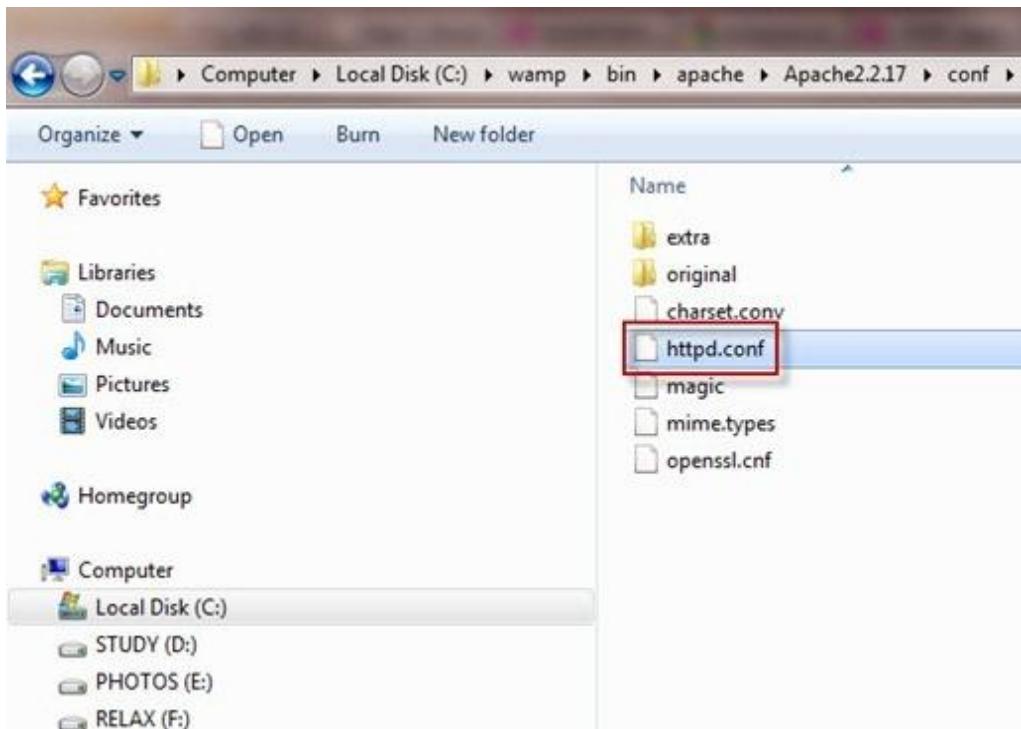


**Hình 4-2-10: Bước 9 - quá trình cài đặt WampServer2**

Do Apache và IIS sử dụng cùng cổng 80 nên để tránh xung đột, bạn có thể tắt dịch vụ IIS trên máy tính của bạn hoặc thay đổi cổng mặc định 80 cho Apache.

Để tắt IIS, bạn vào Start → Settings → Control Panel → Administrative Tools → Internet Information Services → sau đó bạn tìm đến mục Default Web site trong thư mục Web Sites trên máy của bạn, chọn Stop.

Để thay đổi cổng mặc định của Apache trong WampServer2 sang giá trị khác, ví dụ: 70, bạn tìm tới file **httpd.conf** trong thư mục C:\wamp\bin\apache\Apache2.2.17\conf



**Hình 4-2-11: Đổi cổng cho Apache**

Bạn dùng một chương trình soạn thảo (VD: [notepad++](#)) để mở file **httpd.conf** sau đó bạn nhấn **Ctr + F** và tìm kiếm với giá trị 80, bạn sẽ tìm thấy được dòng sau:

```
#Listen 12.34.56.78:80Listen 80
```

Bạn thay số **80** sang 1 giá trị khác, VD: **70** sau đó lưu file **httpd.conf** lại.

Để việc thay đổi cổng cho Apache có hiệu lực bạn hãy tiến hành khởi động lại WampServer2, bạn thực hiện việc này bằng cách click vào biểu tượng WampServer2 trên thanh System Tray sau đó chọn **Restart All Services** trong menu hiện lên.



**Hình 4-2-12: Khởi động lại các dịch vụ trong WampServer2**

### 4.2.3. Sử dụng WampServer 2

Sau khi thay đổi cổng cho Apache và khởi động các dịch vụ trong gói WampServer2, bạn truy cập vào trình duyệt Firefox (*trình duyệt đã chọn là mặc định của WampServer 2*) và gõ vào đường URL sau <http://localhost:70> để truy cập vào WampServer2

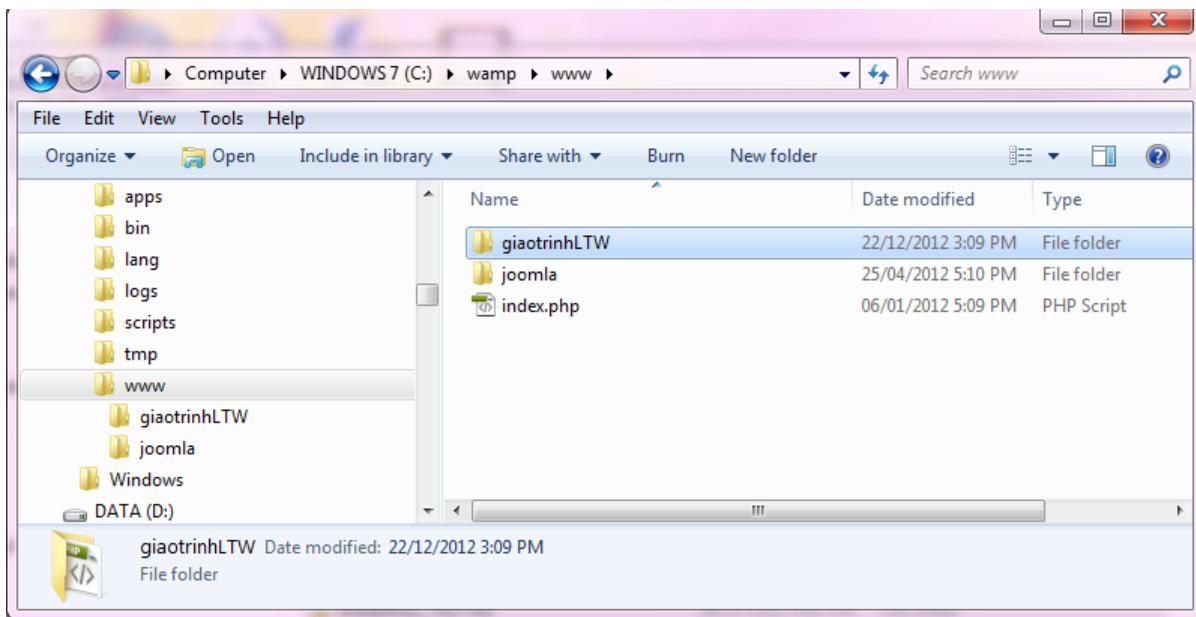
The screenshot shows the WampServer 2.1 control panel. At the top, it displays the WampServer logo and the version information: Version 2.1 Version Française. Below this, there's a section titled "Server Configuration" with details about Apache Version (2.2.17), PHP Version (5.3.4), and MySQL Version (5.1.33). A "Loaded Extensions" table lists numerous PHP extensions. On the left, there are three sections: "Tools" containing "phpinfo()", "phpmyadmin", and "Your Projects" with "wordpress"; "Your Aliases" containing "phpmyadmin", "sqlBuddy", and "webgrind". The "phpmyadmin" link in the Tools section is highlighted with a red box.

**Hình 4-2-13: Giao diện quản trị WebServer bằng WampServer2**

Tại màn hình này bạn sẽ được cung cấp các thông tin và công cụ giúp quản trị WampServer2 như *phiên bản Apache, PHP*, các công cụ *phpMyAdmin, sqlBuddy*, bạn cũng

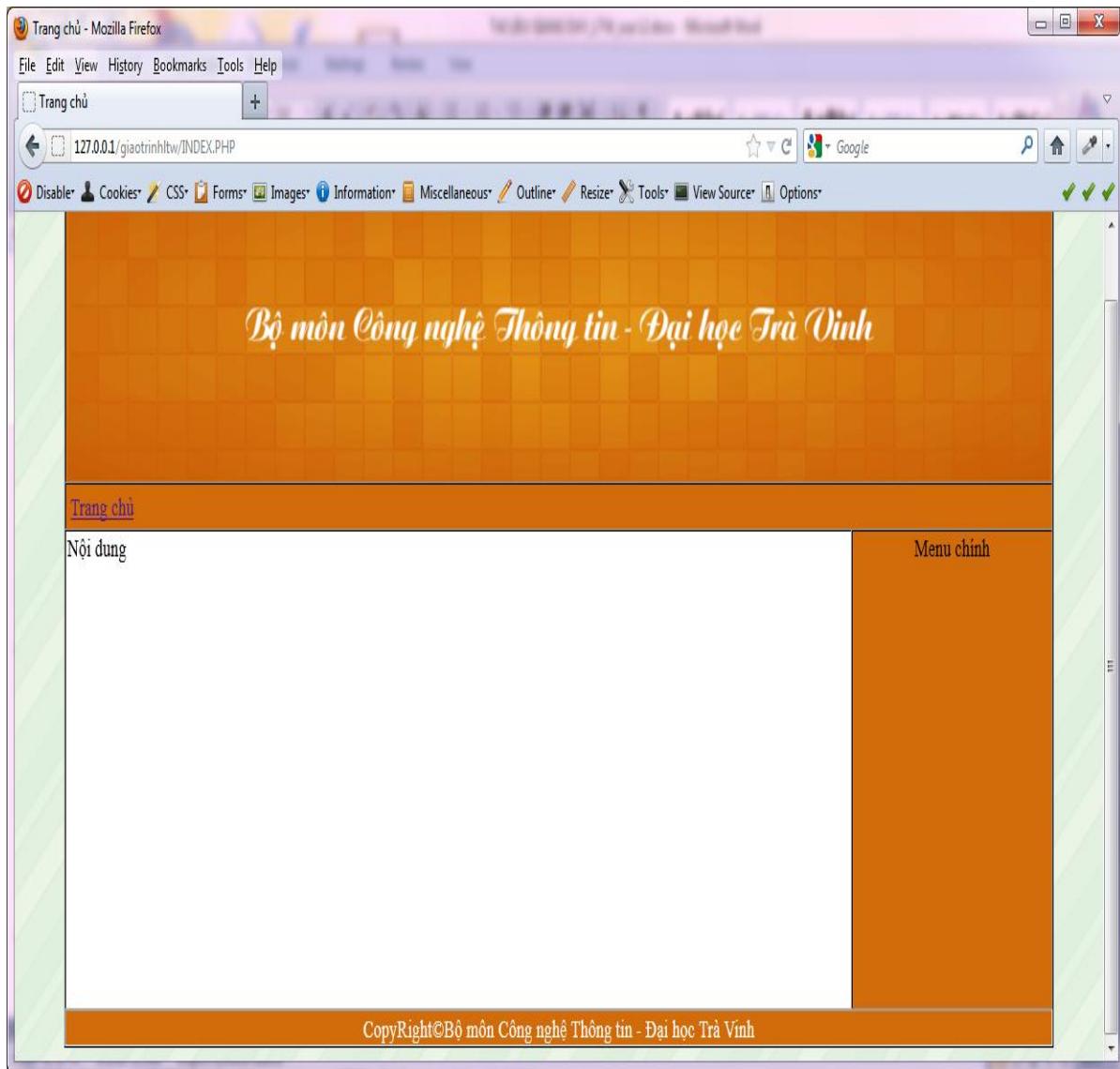
sẽ thấy ứng dụng PHP của bạn xuất hiện trong mục Your Projects.

Để thực thi ứng dụng PHP trên Wamp bạn phải lưu các trang web vào thư mục **www** theo đường dẫn <C:\wamp\www>. Bạn có thể tạo cho mình một thư mục để chứa Website, ví dụ như hình bên dưới, chúng tôi tạo một thư mục chứa Website tên là **giaotrinhLTW**



**Hình 4-2-14: Tạo một thư mục chứa Website với tên là `giaotrinhLTW`**

Như đã trình bày ở mục 1.2.1 của bài này, chúng ta cần tạo một trang **index.php** làm trang chủ cho Website. Sau đó bạn mở trình duyệt, gõ vào đường dẫn <http://localhost:70/tenthumucWeb/> (thay số 70 thành số cổng mà bạn đã đổi nếu ban đầu bạn chọn cách đổi cổng cho Apache; hoặc bỏ :70 trong địa chỉ trên nếu bạn chọn cách tắt IIS) hoặc bạn click chuột trái vào biểu tượng của Wamp trên thanh System Tray → chọn Localhost → chọn thư mục chứa web vừa tạo. Một điều bạn cần chú ý nữa là localhost trên máy tính của bạn sẽ có địa chỉ IP là 127.0.0.1. Do đó thay vì bạn gõ <http://localhost:70/tenthumucWeb/>, bạn có thể gõ thành <http://127.0.0.1:70/tenthumucWeb/> với 70 cũng là cổng Apache đang sử dụng. Tên thư mục, tên tập tin web không phân biệt HOA / thường trên thanh đường dẫn của trình duyệt.



**Hình 4-2-15: Duyệt thử trang web trên WampServer**

❖ Câu hỏi (bài tập) củng cố:

1. Cài đặt wampserver.
2. Sau khi cài đặt xong, sinh viên hãy tạo một thư mục có tên là **hotenSV** và tạo trang **index.php** để có được giao diện như hình 6-2-15

# CHƯƠNG 5

## PHP CƠ BẢN

### BÀI 1

#### **CÁC QUY ƯỚC VÀ CÁCH KHAI BÁO**

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Hiểu rõ được các quy ước khi sử dụng ngôn ngữ PHP
- Khai báo và sử dụng biến trong ngôn ngữ PHP

#### 5.1.1. Quy ước trong PHP

Mã lệnh PHP được chèn vào vị trí nào trong trang HTML sẽ được biên dịch và hiển thị kết quả tại vị trí chèn mã lệnh. Có thể gọi mã lệnh PHP bằng nhiều cách. Một đoạn mã lệnh PHP có thể được đặt trong các cặp thẻ sau:

Thẻ mở	Thẻ đóng
<?	?>
<?php	?>
<script language="php">	</script>

Trong PHP phân biệt chữ hoa và chữ thường và tất cả các câu lệnh php đều cách nhau bởi dấu chấm phẩy (;). PHP không phân biệt khoảng trắng, Tab, xuống dòng.

Để ghép chuỗi trong PHP ta dùng dấu chấm (.)

Việc ghi chú lại ý nghĩa của dòng lệnh trong các ngôn ngữ lập trình là một vấn đề hết sức cần thiết. Ghi chú lại giúp chúng ta dễ dàng biết được chúng ta viết lệnh này dùng để làm gì phục vụ cho công việc gì. Trong PHP có 3 cách ghi chú:

// Đây là ghi chú một dòng

# Đây là ghi chú một dòng

```
/* Đây là ghi
chú nhiều dòng */
```

Ví dụ:

Mã lệnh	Kết quả
<pre>&lt;?php // Khoi tao bien A \$A=5; \$a=10; /*Hien thi bien A va bien a ra man hinh */ echo "A= ".\$A." a= ".\$a; ?&gt;</pre>	A= 5 a= 10

### 5.1.2. Cách khai báo biến

Biến trong PHP không cần khai báo kiểu dữ liệu, kiểu của biến sẽ được xác định tại thời điểm gán trị cho biến. Cú pháp khai báo như sau:

\$ten\_bien=giá trị;

Trong đó, tên biến:

Có thể bao gồm các ký tự (A đến Z, a đến z), ký tự số (0...9), \_, \$

Không được khai báo biến bắt đầu bằng ký số (0...9)

Phân biệt chữ HOA – chữ thường

Ví dụ: khai báo biến a có giá trị là 5 (\$a=5;).

Tên biến động nghĩa là có thể đặt tên biến theo giá trị lưu trong một biến khác.

Ví dụ:

Mã lệnh	Kết quả
<pre>&lt;?php \$ly=5; \$mon= "ly"; echo "Diem mon ". \$mon. " la ". \$\$mon; ?&gt;</pre>	Diem mon ly la 5

Có thể dùng từ khóa define để khai báo hằng. Ví dụ để khai báo một hằng số tên hang1 mang giá trị là 10. Ta khai báo như sau:

```
define("hang1",10);
```

### 5.1.3. Kiểu dữ liệu trong PHP

Biến trong PHP có thể lưu bất kỳ kiểu dữ liệu nào.

Các kiểu dữ liệu gồm: boolean (bool), integer (int), double (float, real), string, array, object.

### 5.1.4. Kiểu chuỗi – string

Toán tử nối chuỗi: dấu chấm (.)

Ví dụ: \$s="xin chao"." ban"; // \$s="xin chao ban";

Phân biện dấu nháy đơn và nháy kép: Các biến nằm trong chuỗi với dấu nháy kép sẽ được tính toán giá trị.

Ví dụ:

```
$t="Tam";
echo ' Chao $t'; // Chao $t
echo " Chao $t"; // Chao Tam
```

Một số hàm xử lý chuỗi:

Hàm strlen(): được dùng để trả về chiều dài của một chuỗi.

Ví dụ:

Mã lệnh	Kết quả
<?php     \$s= "Hello world!!!";     echo strlen(\$s); ?>	14

Hàm strpos(): được sử dụng để tìm kiếm một ký tự trong chuỗi. Nếu hàm tìm thấy sẽ trả về vị trí của ký tự, ngược lại sẽ trả về false.

Ví dụ:

Mã lệnh	Kết quả
<?php \$s= "Hello world!!!"; echo strpos(\$s, "world"); ?>	6

### **Một số hàm xử lý ngày:**

#### Lấy thời gian:

Để tạo một nhãn thời gian, bạn sử dụng hàm **time** hoặc **mkttime**. Hàm time tạo nhãn thời gian cho ngày tháng giờ hiện tại, còn hàm mkttime tạo nhãn thời gian cho giá trị mà bạn truyền vào. Những hàm này đều trả về nhãn thời gian. Để định dạng lại nhãn thời gian, bạn có thể sử dụng hàm date như được hướng dẫn ở minh họa phía trên.

Khi sử dụng hàm mkttime, các tham số không bắt buộc. Vì vậy, bạn có thể bỏ qua các tham số (tính từ tham số cuối) và chỉ làm việc với các tham số đầu. Nếu bỏ qua tham số, hàm mkttime sử dụng giá trị hiện tại. Ví dụ, tham số cuối cùng định nghĩa năm, vì thế nếu bỏ qua tham số này, hàm mkttime sẽ sử dụng năm hiện tại cho nhãn thời gian. Tương tự, nếu bỏ qua 2 tham số cuối, hàm mkttime sử dụng năm và tháng hiện tại.

Tên hàm	Mô tả
time()	Trả về ngày giờ hiện tại dưới dạng nhãn thời gian
mkttime{[\$h[, \$m[, \$s[, \$M[, \$D[, \$Y]]]]]]}	Trả về nhãn thời gian dựa vào ngày tháng và thời gian truyền vào. Nếu bất cứ phần nào bị bỏ qua thì nó sẽ lấy giá trị của ngày giờ hiện tại.
checkdate(\$M, \$D, \$Y)	Trả về true nếu tháng, ngày, năm là hợp lệ.
getdate([\$ts])	Nếu bỏ qua tham số, hàm lấy ngày giờ hiện tại.

Bạn có thể sử dụng hàm strtotime để sinh nhãn thời gian đối với hầu hết các chuỗi

định ngày tháng. Để làm điều này, bạn có thể truyền vào hàm một chuỗi định ngày tháng, thời gian, hay cả ngày tháng và thời gian. Nó hoạt động tốt với các chuỗi lưu trữ ngày giờ theo định dạng hay dùng nhất. Kiểu chuỗi này còn được biết đến với tên gọi *mẫu tuyệt đối*.

Tên hàm	Mô tả
<code>strtotime(\$str[, \$ts])</code>	Trả về nhãn thời gian cho chuỗi truyền vào. Mặc định, hàm này sẽ làm việc tương đối với ngày giờ hiện tại. Tuy nhiên, nếu bạn cung cấp thêm tam số \$ts, hàm sẽ xử lý tương đối với nhãn thời gian truyền vào này. Hàm có khả năng hiểu đa phần định dạng ngày giờ và mô tả thời gian tương đối.

### Các kiểu mẫu được dùng với strtotime

Loại	Mô tả
Tuyệt đối	<p>Chuỗi định dạng định nghĩa ngày tháng và thời gian. Nếu ngày không có giờ thì giờ sẽ được khởi tạo lúc nửa đêm. Nếu một số bộ phận của ngày (ví dụ, chỉ có tháng và ngày) được truyền vào thì các bộ phận khác mặc định hiểu không đổi.</p> <p>Giờ không có ngày sẽ không đổi ngày. Một số bộ phận của giờ (ví dụ chỉ có giờ và phút) sẽ thiết lập những bộ phận khác về 0.</p>
Tương đối	Khoảng cách được thêm vào hoặc trừ bớt đi từ ngày giờ gốc. Một vài ngày tương đối sẽ đặt giờ về lúc nửa đêm.

### Dịnh dạng thời gian:

```
$date=date ("format", $timestamp);
```

Ý nghĩa: Trả về chuỗi mô tả ngày tháng đã được định dạng theo chuỗi định dạng \$format. Mặc định, hàm làm việc với ngày giờ hiện tại. Tuy nhiên, bạn có thể dùng biến thứ hai để định nghĩa một nhãn thời gian cho bất kỳ ngày giờ nào.

Ví dụ: `$today=date ("Y/m/d"); $now=date ("h:i:s");`

### Mã định dạng thông dụng của hàm date

Ký tự	Mô tả	Từ	Tới
D	Ngày trong tuần – 3 chữ cái	Mon	Sun
l	Ngày trong tuần – đủ chữ	Monday	Sunday
w	Ngày trong tuần (với 0=Sunday)	0	6
z	Ngày trong năm	0	365
S	Số thứ tự của ngày trong tháng (st, nd, rd, th)		
n	Tháng – không có số 0 ở đầu	1	12
m	Tháng – có 0 ở đầu	01	12
M	Tháng – 3 chữ cái đầu	Jan	Dec
F	Tháng – đủ chữ	January	December
j	Ngày trong tháng – không có số 0 ở đầu	1	31
d	Ngày trong tháng – có 0 ở đầu	01	31
y	Năm – 2 chữ số	10	
Y	Năm – 4 chữ số	2010	
L	Năm nhuận (1) hoặc năm thường (0)	0	1
g	Giờ - định dạng 12-giờ, không có 0 đầu	1	12
G	Giờ - định dạng 24-giờ, không có 0 ở đầu	0	23
h	Giờ - định dạng 12-giờ, có 0 ở đầu	01	12
H	Giờ - định dạng 24-giờ, có 0 ở đầu	00	23

Ký tự	Mô tả	Từ	Tới
i	Phút – có 0	00	59
s	Giây – có 0	00	59
a	am/pm – viết thường	am	pm
A	AM/PM – viết hoa	AM	PM
T	Viết tắt cho vùng thời gian	EST	
U	số giấy kể từ kỷ nguyên Unix	-2,147,483,648	2,147,483,647

### 5.1.5. Các phép toán trong PHP

#### 5.1.5.1. Các phép toán số học

Phép toán	Mô tả	Ví dụ	Kết quả
+	Cộng	$x=2$ $x+2$	4
-	Trừ	$x=2$ $5-x$	3
*	Nhân	$x=4$ $x*5$	20
/	Chia	$15/5$ $5/2$	3 2.5
%	Chia lấy phần dư	$5\%2$ $10\%8$ $10\%2$	1 2 0
++	Tăng 1 đơn vị	$x=5$ $x++$	$x=6$
--	Giảm 1 đơn vị	$x=5$ $x--$	$x=4$

### 5.1.5.2. Phép gán

Phép gán	Ví dụ	Cách viết đầy đủ
=	x=y	x=y
+=	x+=y	x=x+y
-=	x-=y	x=x-y
*=	x*=y	x=x*y
/=	x/=y	x=x/y
.=	x.=y	x=x.y
%=	x%-=y	x=x% y

### 5.1.5.3. Các phép toán so sánh

Phép toán	Mô tả	Ví dụ
==	Bằng	5==8 trả về giá trị false
!=	Không bằng (khác)	5!=8 trả về giá trị true
<>	Không bằng (khác)	5<>8 trả về true
>	Lớn hơn	5>8 trả về false
<	Nhỏ hơn	5<8 trả về true
>=	Lớn hơn hoặc bằng	5>=8 trả về false
<=	Nhỏ hơn hoặc bằng	5<=8 trả về true

#### 5.1.5.4. Phép toán logic

Phép toán	Mô tả	Ví dụ
&&	And	x=6 y=3 $(x < 10 \&\& y > 1)$ trả về giá trị true
	Or	x=6 y=3 $(x==5 \mid\mid y==5)$ trả về giá trị false
!	Not	x=6 y=3 $!(x==y)$ trả về giá trị true

❖ Câu hỏi (bài tập) củng cố:

1. Hiển thị dòng chữ “Xin chào PHP” bằng PHP
2. Cho a=5,b=4. Hiển thị kết quả a+b ra màn hình
3. Hiển thị ngày tháng năm hiện tại ra màn hình

## BÀI 2

### **MỆNH ĐỀ ĐIỀU KIỆN VÀ CẤU TRÚC LẶP**

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Sử dụng được các câu lệnh điều kiện: if, switch
- Sử dụng được các cấu trúc lặp: while, for

#### **5.2.1. Mệnh đề điều kiện**

Mệnh đề điều kiện được sử dụng để biểu diễn các hành động khác nhau dựa trên các điều kiện.

- Mệnh đề if: sử dụng mệnh đề này để thực thi một số mã lệnh chỉ khi điều kiện đưa ra là đúng
- Mệnh đề if...else: sử dụng mệnh đề này để thực thi một số mã lệnh khi điều kiện đúng và thực thi một số mã lệnh khác khi điều kiện là sai.
- Mệnh đề if...elseif....else: sử dụng mệnh đề này để chọn một trong nhiều khối mã lệnh để thực thi.
- Mệnh đề switch: Sử dụng mệnh đề này để chọn một trong nhiều khối mã lệnh để thực thi.

Lưu ý:

Nếu mã lệnh thực thi có từ 2 câu lệnh trở lên, chúng phải được đặt trong cặp dấu {...}

##### **5.2.1.1. Mệnh đề if**

Cú pháp:

```
if(điều kiện) {
    // các mã lệnh thực thi khi điều kiện đúng;
}
```

##### **5.2.1.2. Mệnh đề if... else**

Cú pháp:

```
if(điều kiện)
    { // các mã lệnh thực thi khi điều kiện đúng}
else
    { // các mã lệnh thực thi khi điều kiện sai}
```

### **5.2.1.3. Mệnh đề if...elseif...else**

Cú pháp:

```
if(điều kiện)
    {// các mã lệnh thực thi khi điều kiện đúng}
elseif(điều kiện)
    {// các mã lệnh thực thi khi điều kiện đúng}
else
    {// các mã lệnh thực thi khi điều kiện sai}
```

### **5.2.1.4. Mệnh đề switch**

Để khắc phục sự dài dòng trong câu lệnh if, PHP cho phép lập trình viên sử dụng câu lệnh switch. Câu lệnh này sẽ được sử dụng để kiểm tra khi muốn xem xét qua nhiều giá trị của một biến.

Cú pháp:

```
switch(bien) {
    case gt1:
        // các câu lệnh; break;
    case gt2:
        // các câu lệnh; break;
    ...
    case n:
        // các câu lệnh; break;
    default: // các câu lệnh;
}
```

## **5.2.2. Cấu trúc lặp**

### **5.2.2.1. Cấu trúc while**

While là câu lệnh lặp không biết trước số lần lặp tức là nó lặp một đoạn lệnh nào đó cho đến khi thỏa mãn điều kiện đặt ra.

Cú pháp:

```
while(điều kiện) {
    // các mã lệnh thực thi
}
```

Ví dụ 5-2-1:

Ví dụ	Kết quả
<pre>&lt;html&gt; &lt;body&gt; &lt;?php     \$num=array(2,3,5,7,11,13,17,19);     \$i=0;     while(\$i&lt;8)     {         echo \$num[\$i]. " ";         \$i++;     } ?&gt; &lt;/body&gt; &lt;/html&gt;</pre>	2 3 5 7 11 13 17 19

#### 5.2.2.2. Cấu trúc do...while

do... while là câu lệnh lặp không biết trước số lần lặp tức là nó lặp 1 đoạn lệnh nào đó cho đến khi thỏa mãn điều kiện đặt ra. So với cấu trúc while thì cấu trúc do...while sẽ thực hiện ít nhất một lần lặp.

Cú pháp:

```
do
{
    // các mã lệnh thực thi
}while(điều kiện);
```

Ví dụ 5-2-2:

Ví dụ	Kết quả
<pre>&lt;html&gt; &lt;body&gt; &lt;?php     \$num=array(2,3,5,7,11,13,17,19);     \$i=0;     do     {         echo \$num[\$i]. " ";         \$i++;     } while(\$i&lt;8); ?&gt; &lt;/body&gt; &lt;/html&gt;</pre>	2 3 5 7 11 13 17 19

### 5.2.2.3. Cấu trúc for

for là câu lệnh lặp biết trước số lần lặp tức là nó lặp 1 đoạn lệnh nào đó cho đến khi thỏa mãn điều kiện đặt ra.

Cú pháp:

```
for(biểu thức 1; biểu thức 2; biểu thức 3) {
    // các mã lệnh thực thi
}
```

Trong đó:

- Biểu thức 1: thường được sử dụng để khởi tạo biến đếm, biến điều khiển vòng lặp, ...
- Biểu thức 2: là biểu thức điều kiện, nếu biểu thức này được đánh giá là TRUE thì vòng lặp sẽ tiếp tục thực hiện, ngược lại sẽ dừng khi biểu thức được đánh giá là FALSE.
- Biểu thức 3: thường được sử dụng để điều khiển vòng lặp, tăng hoặc giảm biến điều khiển, ... Biểu thức này được thực thi ở cuối mỗi vòng lặp.

Lưu ý: Trong 3 biểu thức trên nếu ở mỗi biểu thức có hơn 1 biểu thức con thì giữa các biểu thức con được phân cách bởi dấu phẩy.

Ví dụ 5-2-3: Tính tổng n số nguyên dương đầu tiên (với n=5)

```

<html>
<body>
<?php
$n=5;
    for($i=1, $s=0; $i<=$n;$i++)
        $s+=$i;
    echo "Tong cua ".$n." so nguyen duong dau tien la:".$s;
?>
</body>
</html>

```

Kết quả:

Tong cua 5 so nguyen duong dau tien la 15

#### **5.2.2.4. Cấu trúc foreach**

Từ PHP 4 vòng lặp foreach chỉ được sử dụng để lặp thông qua một mảng (PHP 5 hỗ trợ cho cả đối tượng và tham chiếu). Chúng ta sẽ tìm hiểu sâu hơn về mảng ở BÀI 3.

Cú pháp:

```

foreach($tên_mảng as $value) {
    // các mã lệnh thực thi
}

```

Mỗi lần lặp được thực hiện thì giá trị của phần tử hiện tại trong mảng sẽ được gán bằng giá trị của \$value và con trỏ của mảng dịch chuyển đi một vì vậy trong lần lặp tiếp theo sẽ là giá trị của phần tử tiếp theo trong mảng.

Ví dụ 5-2-4: Tính tổng các phần tử trong mảng

```

<html>
<body>
<?php
    $num=array(2, 4, 6, 8);
    $s=0;
    foreach($num as $value)
        $s+=$value;
    echo "Tong cua cac phan tu trong mang la: ".$s;
?>
</body>
</html>

```

Kết quả

Tong cua cac phan tu trong mang la: 20

❖ Câu hỏi (bài tập) củng cố:

Cho a trong khoảng 1 đến 10.

1. Hãy viết chương trình đọc số a bằng hàm if
2. Hãy viết chương trình đọc số a bằng hàm switch
3. Hãy tính tổng các số nhỏ hơn a bằng hàm for
4. Hãy tính tổng các số nhỏ hơn bằng a bằng hàm while
5. Hãy tính tổng các số nhỏ hơn a bằng hàm do... while
6. Kiểm tra một số n nguyên dương cho trước có phải là số nguyên tố hay không?
7. Trong nhân gian có câu đố: vừa gà vừa chó bó lại cho tròn đếm đủ 36 con 100 chân chẵn. Hỏi có bao nhiêu con gà, bao nhiêu con chó?
8. Theo một nhóm người cho rằng, biển số xe gắn máy đẹp là biển số có 4. Các số phải giống nhau hoặc tổng 4 số cộng lại có số cuối cùng bằng 9.

Ví dụ: Biển số xe 6666 là biển số đẹp. Biển số xe 1279 là biển số đẹp (vì  $1+2+7+9=19$ )

## BÀI 3

### **MẢNG VÀ HÀM TRONG PHP**

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Biết cách sử dụng mảng
- Biết khai báo và sử dụng hàm

#### **5.3.1. Mảng trong PHP**

Mảng có thể chứa nhiều thông tin riêng biệt, mảng giống như một danh sách các giá trị, mỗi giá trị là một biến, một chuỗi, một số, thậm chí là một mảng khác.

Mảng được câu tạo dưới dạng một chuỗi các cặp khóa giá trị, mỗi mục trong danh sách đều có một khóa (hoặc chỉ số index gắn liền với chúng).

Mảng trong PHP cũng phải tuân thủ theo quy tắc đặt tên của biến, để nhận giá trị từ mảng ta cần tham chiếu đến tên mảng và khóa đặt trong dấu ngoặc vuông:

Ví dụ: echo \$list [2] hoặc echo \$nation ['VN'];

##### **5.3.1.1. Mảng sử dụng các chỉ số**

Loại này dùng các con số làm "khóa".

Ví dụ: Mảng \$list

Khoa	Giá trị
1	Low
2	Middle
3	High

##### **5.3.1.2. Mảng sử dụng các chuỗi làm khóa**

Ví dụ: Mảng \$nation

Khoa	Giá trị

Khoá	Giá trị
VN	Việt Nam
US	United States
RU	Russia
CH	China

### 5.3.2. Các mảng siêu toàn cục

Mảng rất mạnh và hữu dụng trong PHP, có các mảng mặc định như các mảng siêu toàn cục: \$\_GET; \$\_POST, \$\_COOKIE...biến \$\_GET là nơi chứa toàn bộ giá trị biến được gửi theo phương thức GET (không nhất thiết là trong biểu mẫu).

VD: Ta có đường link như sau: <http://index.php?id=2&name=product>

Khi có link này chúng ta có thể nhận được 02 biến sau:

`$_GET[id] = 2 và $_GET[name]=product`

Đây là mảng \$\_GET có khóa là tên biến id, name và có giá trị là 2, product. Thông thường các biến này được gửi lên từ một FORM nào đó hoặc đường link được lập trình sẵn.

### 5.3.3. Tạo mảng: Có nhiều cách để tạo mảng:

#### 5.3.3.1. Dùng hàm array()

Ví dụ 5-3-1:

```
<?php
    $mang = array ('VN' => 'Việt Nam', 'LA' => 'Lào');
    echo " VN là tên viết tắt của {$mang['VN']} ";
?>
```

Nếu khóa đặt trước là dạng số thì các giá trị sau được tự động gán các khóa số tiếp tục

Ví dụ 5-3-2:

```
<?php
    $mang = array (1 => 'Việt Nam', 'Lào');
    echo " LA là tên viết tắt của {$mang[2]} ";
?>
```

### 5.3.3.2. Tạo mảng số với hàm range()

Ví dụ 5-3-3:

```
<?php
    $mangso = range (1, 10); //cũng có thể dùng cho chữ cái
        echo $mangso[3]; //câu lệnh này in ra giá trị là 4, là
    số đứng vị trí thứ 3 trong mảng tính từ 0
?>
```

### 5.3.3.3. Tạo mảng với dữ liệu bất kỳ với vòng lặp

Ví dụ 5-3-4:

```
<?php
for ($i=0;$i<=100;$i++) {
    $mang []=$i;
}
?>
```

Có thể lặp bắt cứ dữ liệu j (số, chuỗi, dữ liệu từ Database...) và thêm phần tử vào mảng

### 5.3.4. Truy xuất mảng

Chúng ta có thể dùng hàm foreach() để truy xuất các phần tử mảng (hàm này làm việc dựa trên bản sao của mảng nên sẽ không ảnh hưởng đến dữ liệu của mảng nhưng sẽ ảnh hưởng đến bộ nhớ nếu làm việc với mảng lớn):

Ví dụ 5-3-5:

```
<?php
$months = array (1 => 'January', 'Feburary', 'March', 'April',
'May', 'June', 'July', 'August', 'September', 'October',
'November', 'December');
$days = range (1, 31);
$years = range (2000, 2020);
echo '<select name="month">';
foreach ($months as $key => $value) {
    echo "<option value=\"$key\">$value</option>\n";
}
echo '</select><select name="day">';
foreach ($days as $value) {
    echo "<option value=\"$value\">$value</option>\n";
}
echo '</select><select name="years">';
foreach ($years as $value) {
```

```

        echo "<option value=\"$value\">$value</option>\n";
    }
echo '</select>';
?>

```

#### **Chú thích:**

Chúng ta lần lượt tạo ra các mảng Ngày Tháng Năm và dùng mã HTML <option> tạo ra các lựa chọn xổ xuống. Vòng lặp foreach sẽ liệt kê ra các giá trị của mảng.

#### **Kết quả sẽ cho ta các danh sách về ngày tháng năm**

#### **5.3.5. Các hàm thông dụng xử lý mảng**

Hàm count(), sizeof() : trả về kích thước của mảng

Các hàm “di chuyển” trong mảng :

- + current()/end() : trả về phần tử đầu tiên/ cuối cùng của mảng
- + prev() / next() : trả về phần tử liền trước / kế tiếp

Các hàm sắp xếp mảng :

- + sort() : sắp xếp mảng theo value của phần tử
- + ksort():sắp xếp mảng theo key của phần tử

Kiểm tra một biến có phải là mảng : is\_array() trả về 1 (true) và 0 (false)

#### **5.3.6. Mảng đa chiều**

Hãy cùng xem ví dụ 5-3-6:

```

<?php
$nations = array ('VN' => 'Việt Nam', 'US' => 'United States',
'UK' => 'United Kingdom');
$vn = array('HN' => 'Hà Nội', 'HCM' => 'Hồ Chí Minh');
$n = array ('VN' => $vn, 'US' => @$unitedstates, 'UK' =>
@$unitedkingdom);
echo " Thủ đô của {$n['VN']} là {$n['VN']['HN']} ";
// kết quả sẽ là: Thủ đô của Việt Nam là Hà Nội.
?>

```

### **Chú thích:**

Ta tạo 2 mảng \$nations và \$vietnam. Ta tạo mảng đa chiều bằng cách lấy khóa của mảng làm khóa cho giá trị của mảng đa chiều. Trong trường hợp trên mảng \$nations không liên quan gì đến mảng \$n và \$vietnam cả, mà thực tế là khóa ‘VN’ của mảng \$n sẽ có giá trị là mảng \$vietnam.

### **5.3.7. Chuyển đổi giữa mảng và chuỗi**

Chúng ta có thể chuyển các phần tử của mảng thành kiểu chuỗi và in chúng ra một cách dễ dàng thay cho việc in từng phần tử của mảng:

Ví dụ 5-3-7:

```
<?php
    $months = array (1 => 'January', 'Feburary', 'March',
    'April', 'May', 'June', 'July', 'August',
    'September', 'October', 'November', 'December'); //Mảng chứa các
    tháng trong năm
    $string = implode (' - ', $months); //chuyển đổi mảng trên
    thành chuỗi và in ra với dấu ' - ' phân cách
    echo " Các tháng trong năm $string";
    $array = explode (' - ', $string); //chuyển đổi chuỗi vừa
    có thành mảng và in ra tháng 01
    echo " Tháng đầu tiên trong năm là {$array[1]}";
?
>
```

**Lưu ý:** trong cả 02 trường hợp chuyển đổi trên ta phải chỉ ra dấu phân cách các phần tử để PHP có thể nhận biết .

### **5.3.8. Sắp xếp mảng**

Để sắp xếp mảng chúng ta có hai sự chọn lựa: sắp xếp theo khoá hoặc sắp xếp theo giá trị. Bên dưới là ví dụ minh họa cho thấy một số hàm hỗ trợ của PHP cho việc sắp xếp mảng.

Ví dụ 5-3-8:

```

<?php
echo "Mảng thẻ hiện bảng điểm</br>";
$mark = array ('10' => 'Peter', '05' => 'Tom', '02' =>
'Mary');
foreach ($mark as $key => $value)
{    echo "$key\t$value </br>";}
echo '</pre>';

//Xắp xếp theo thứ tự chữ cái tên
echo "</br> Xắp theo tên </br>";
asort ($mark); //Các khóa vẫn được gắn với giá trị của nó
foreach ($mark as $key => $value)
{
    echo "$key\t$value </br>";}
echo '</pre>';

//Xắp xếp theo thứ tự khóa tăng dần
echo "</br> Xắp theo thứ tự khóa tăng dần </br>";
ksort ($mark); //Các khóa vẫn được gắn với giá trị của nó
foreach ($mark as $key => $value)
{
    echo "$key\t$value </br>";}
echo '</pre>';

//Xắp xếp theo thứ tự khóa giảm dần
echo "</br> Xắp theo thứ tự khóa giảm dần </br>";
krsort ($mark); //Các khóa vẫn được gắn với giá trị của nó
foreach ($mark as $key => $value)
{
    echo "$key\t$value </br>";}
echo '</pre>';

?>

```

### 5.3.9. Hàm người dùng định nghĩa

#### 5.3.9.1. Cú pháp:

```
function tên_hàm([các_tham_số]) {  
    // Khai báo biến riêng  
    // Câu lệnh PHP  
    [ return giá_trị; ]  
}
```

Trong đó, [các\_tham\_số]: các tham số là tùy chọn.

#### 5.3.9.2. Sử dụng hàm

Hàm không có giá trị trả về : tên\_hàm([các\_tham\_số]);

Hàm có giá trị trả về : tham gia vào các biểu thức, gán giá trị cho biến.

Ví dụ 5-3-9: hàm không có giá trị trả về

```
<?php  
$arr = array(hoten=>"Hà Giang",diachi=>"Đà  
 Nẵn",tuoi=>15,phai=>"Nữ");  
  
function display($a){  
    echo "<table border=1>";  
    reset($a); // trả về phần tử đầu tiên của mảng $a;  
    $value=current($a); // phần tử hiện thời của mảng $a  
    $str="";  
    while($value){  
        $str.= "<tr><td>".$value."</td></tr>";  
        $value=next($a); // phần tử kế tiếp của mảng $a  
    }  
    $str.= "</table>";  
    echo $str;  
}  
display($arr);  
?>
```

Ví dụ 5-3-10: hàm có giá trị trả về

```
<?php  
$arr=array(hoten=>"Hà Giang",diachi=>"Đà Nẵng");  
function is_hoten($mang) {  
    // trả về value của phần tử có key là hoten  
    while(list($k,$v)=each($mang)) {  
        if($k=="hoten") {return $v;exit;}  
    }  
}  
echo is_hoten($arr);  
?>
```

#### 5.3.9.3. *Tầm vực của biến trong trang PHP*

Khi khai báo hoặc sử dụng biến trong bất kỳ ngôn ngữ lập trình nào, chúng ta cần phải xác định được thời gian sống của biến đó. Một biến có giá trị toàn cục khi nó được khai báo ngoài phạm vi các hàm, ngược lại các biến khai báo bên trong hàm là biến có giá trị cục bộ.

Tuy nhiên trong PHP có trường hợp ngoại lệ là khi biến khai báo bên trong hàm nhưng có khai báo thêm từ khoá global thì biến đó được xem là biến có giá trị toàn cục. Biến \$\_SESSION cũng được xem là biến có giá trị toàn cục, có thể sử dụng cho mọi trang trong phiên làm việc.

```
function tên_hàm($x) {  
    global $total; // bây giờ $total trở thành biến có  
    // giá trị toàn cục  
    // các lệnh PHP  
}
```

#### 5.3.9.4. *Tham trị và tham biến*

- Các ví dụ hàm trong mục 2.3.10.2 là hàm được gọi với tham trị.
- Hàm sau đây sẽ được gọi với *tham biến*:

```
function tên_hàm(&$y) {  
    // các lệnh PHP
```

}

### **Vậy:**

Tham trị: Truyền một hoặc nhiều biến vào hàm để sử dụng nhưng khi kết thúc hàm thì giá trị các biến vẫn không thay đổi.

Tham biến: Truyền một hoặc nhiều biến vào hàm để sử dụng, sau khi kết thúc hàm thì giá trị các biến sẽ bị thay đổi.

### **5.3.10. Sử dụng lại mã PHP**

- Sử dụng hàm include(), require() để chèn tệp tin PHP, HTML vào một trang PHP khác. Ngoài ra còn có: include\_once(), require\_once(), tuy nhiên một số lập trình viên khuyên không lạm dụng, vì làm chậm thực thi script.

- Phân biệt include() và require()

+ include("tên\_tệp") : nếu tên\_tệp không tồn tại thì PHP thông báo lỗi và vẫn tiếp tục thực thi script còn lại.

+ require("tên\_tệp") : nếu tên\_tệp không tồn tại thì PHP thông báo lỗi và dừng thực thi script còn lại.

Sử dụng include(), require() là một cách tái sử dụng đoạn mã đã viết.

Ví dụ 5-3-11:

Trang **Index.php** bao gồm 3 khối:

header.inc : gồm các khai báo <html>...<body>

menu.php : gồm các liên kết

footer.php: gồm các thông tin về website và </body></html>

**Nội dung trang index.php như sau:**

```
<?php
    require ("header.inc");
    require ("menu.php");
    require ("footer.php");
?>
```

**Nội dung trang header.inc**

```
<html><head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8">
<style>
    body{margin:10 10 10 60; font: normal 18pt Arial;
color:navy}
    // khai báo để DEMO hiển thị rõ ràng khi trình bày trên
LCD Projector!
</style>
<title>My Page</title>
</head><body> <H2> Ví dụ về include()</H2>
<?php
    echo "<div>Hôm nay:" . date("d-m-Y") . "</div><br>";
?>
```

### Nội dung trang footer.php

```
<?php
    $copyright=<hr /><div class=ft>Copyright by BMCNTT- năm
".date("Y");
    $copyright.= "<br />BMCNTT-TRƯỜNG ĐẠI HỌC TRÀ
VINH</div>";
    // năm hiện thời tự động cập nhật
    echo $copyright;
?>
</body>
</html>
```

### Nội dung trang menu.php

```
<?php  
    $strMenu = "";  
    $strMenu .= "<a href='#'>Trang chu</a><br />";  
    $strMenu .= "<a href='#'>Array</a><br />";  
    $strMenu .= "<a href='#'>String</a><br />";  
    $strMenu .= "<a href='#'>Post, Get</a><br />";  
    $strMenu .= "<a href='#'>Cookie, Session</a><br />";  
    echo $strMenu;  
?>
```

Kết quả hiển thị như màn hình bên dưới:



## Ví dụ về include()

Hôm nay: 21-04-2013

[Trang chu](#)

[Array](#)

[String](#)

[Post, Get](#)

[Cookie, Session](#)

Copyright by BMCNTT2013  
BMCNTT-TRƯỜNG ĐẠI HỌC TRÀ VINH

Hình 5-3-1: Sử dụng lại mã PHP

### 5.3.11. File trong PHP

Việc sử dụng file một cách thành thạo sẽ giúp chúng ta dễ dàng vận hành các ứng dụng mang quy mô vừa và nhỏ như: website nhiều ngôn ngữ, bộ běm...

Khi làm việc trên file, chúng ta cần thực hiện 3 bước cơ bản:

- Bước 1: Mở file

- Bước 2: Thực hiện các thao tác trên file
- Bước 3: Đóng file

### **5.3.11.1.Mở file**

Cú pháp:

```
fopen("đường dẫn", "chế độ")
```

Hàm mở file không thành công sẽ trả về giá trị 0

#### **Các chế độ mở file**

Chế độ	Mô tả
r	Chức năng chỉ đọc và đọc từ đầu file
r+	Chức năng đọc/ghi từ đầu file
w	Chức năng ghi, mở và xoá nội dung của file, hoặc tạo file mới nếu nó không tồn tại.
w+	Chức năng đọc/ghi. Mở và xoá nội dung của file, hoặc tạo file mới nếu nó không tồn tại.
a	Chức năng thêm dữ liệu. Mở và ghi thêm dữ liệu vào cuối trang hoặc sẽ tạo ra file mới nếu không tồn tại.
a+	Chức năng đọc và bổ sung. Lưu giữ lại nội dung của tập tin và ghi thêm vào cuối file.
x	Chức năng chỉ ghi. Tạo ra file mới và trả về giá trị FALSE kèm theo lỗi nếu file đã tồn tại.
x+	Chức năng đọc/ghi. Tạo ra file mớiRead/Write. Tạo ra file mới và trả về giá trị FALSE kèm theo lỗi nếu file đã tồn tại.

Ví dụ 5-3-12:

Trong ví dụ bên dưới, nếu mã lệnh mở file thành công sẽ hiển thị thông báo “Mở file thành công”, ngược lại sẽ thông báo “Không thể mở file”.

```
<html>
<body>
<?php
    $fp=fopen ("hello.txt", "r");
    if ($fp!=0)
        echo "Mở file thành công";
    else
        echo "Không thể mở file";
?>
</body>
</html>
```

Ở ví dụ trên, đoạn mã lệnh PHP chúng ta có thể viết tắt như sau:

```
<html>
<body>
<?php
    $fp=fopen ("hello.txt", "r") or exit("Không thể mở
file");
?>
</body>
</html>
```

### 5.3.11.2. Đóng file

Ví dụ 5-3-13: Đóng tập tin hello.txt đang mở.

```
<html>
<body>
<?php
$fp = fopen("hello.txt", "r");
//Mã lệnh được thực thi khi mở file thành công
fclose($fp);
?>
</body>
</html>
```

### 5.3.11.3. Đọc và ghi file trong PHP

#### a. Đọc một file trong PHP

PHP cho ta nhiều sự lựa chọn trong việc đọc một file. Có nhiều hình thức hỗ trợ nhưng hiện nay hai hình thức phổ biến nhất vẫn là đọc file theo từng dòng và đọc file theo từng ký tự.

### **Đọc file theo từng dòng:**

Cú pháp: fgets(file vừa mở)

Ví dụ 5-3-14:

```
<?php  
    $fp=fopen("test.txt",r) or exit("không thể tìm thấy tập  
    tin để mở");  
    echo fgets($fp);  
    fclose($fp);  
?>
```

### **Đọc file theo từng ký tự:**

Cú pháp: fgetc(file vừa mở)

Ví dụ 5-3-15:

```
<?php  
    $fp=fopen("test.txt",r) or exit("không thể tìm thấy tập  
    tin để mở");  
    echo fgetc($fp);  
    fclose($fp);  
?>
```

### ***b. Ghi một file trong PHP***

Cú pháp cơ bản: fwrite("file vừa mở", "nội dung cần ghi vào file");

Ví dụ 5-3-16: Tạo một tệp test.txt rỗng cùng thư mục với trang php có nội dung bên dưới. Chạy trang php sau đó mở tệp tin test.txt sẽ thấy được chữ “chao ban”.

```
<?php  
    $fp=fopen("test.txt",a) or exit("không thể tìm thấy  
    file để mở");  
    $nd="chao ban";  
    fwrite($fp,$nd);  
?>
```

#### ***5.3.11.4.Kiểm tra kết thúc file***

Ví dụ 5-3-17: Kiểm tra kết thúc file

```
if(feof($fp))  
    echo "Đã cuối file";
```

Sử dụng các hàm fgets() để đọc từng dòng dữ liệu trên file; hàm fgetc() để đọc từng ký tự trên file.

#### Ví dụ 5-3-18: Đọc và hiển thị nội dung tập tin hello.txt

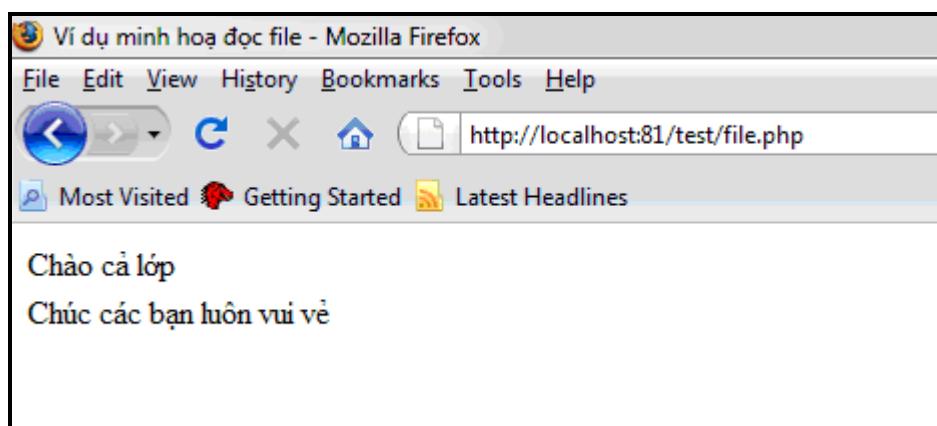
Trong tập tin hello.txt gồm có 2 dòng thông báo:

Chào cả lớp

Chúc các bạn luôn vui vẻ

```
<html>
<body>
<?php
    $fp=fopen("hello.txt", "r");
    if($fp!=0)
    {
        while(!feof($fp))
        {
            echo fgets($fp)."<br>";
        }
    }
    else
        echo "Không thể mở file";
?>
</body>
</html>
```

Kết quả:



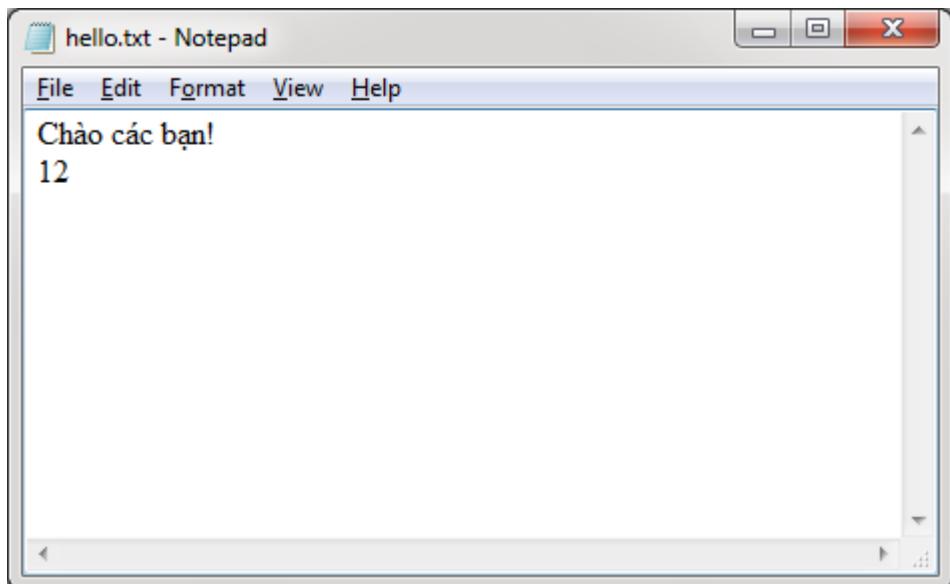
Hình 5-3-2: Đọc file

Ví dụ 5-3-19: Ghi dữ liệu xuống tập tin hello.txt

**Lưu ý:** Khi kiểm tra kết quả thì phải mở file hello.txt để kiểm tra, trên cửa sổ trình duyệt sẽ không hiển thị kết quả, bởi vì chúng ta đang ghi file.

```
<html>
<body>
<?php
    $fp=fopen ("hello.txt", "w+")
        or exit("Không mở được file");
    $n=12;
    if ($fp!=0)
    {
        fwrite($fp, "Chào các bạn!");
        fwrite($fp, "\r\n");
        fwrite($fp, $n);
    }
?>
</body>
</html>
```

Kết quả



**Hình 5-3-3: Ghi dữ liệu xuống tập tin**

### 5.3.12. Lớp đối tượng – Class

#### 5.3.12.1. Cách khai báo

```
class tên_lớp () {  
    var $tên_biến;  
    function tên_hàm () {  
        $this->$tên_biến=giá_trị;  
    }  
}
```

#### 5.3.12.2. Cách sử dụng

```
$bién=new tên_lớp();
```

Ví dụ 5-3-20:

```
class counter()  
{  
    var $count=0;  
    function increment()  
    {  
        $this->count++;  
    }  
}  
$a=new counter(); // khởi tạo  
$a->increment(); // gọi hàm  
echo $a->count; // kết quả là 1;
```

### 5.3.13. Một số hàm thông dụng trong PHP

#### Một số hàm toán học

Tên hàm	Cú pháp	Chức năng	Ví dụ	Kết quả
abs	abs(\$number)	Trả về giá trị tuyệt đối của \$number, nếu \$number là float thì kết quả là float ngược lại là interger	\$abs=abs(-4.2); \$abs=\$abs(-4);	\$abs = 4.2 \$abs=4
ceil	ceil(\$float)	Làm tròn lên số \$float	\$a=ceil(5.1);	\$a=6

Tên hàm	Cú pháp	Chức năng	Ví dụ	Kết quả
floor	floor(\$float)	Làm tròn xuống của \$float	\$a=floor(5.9); \$a=floor(-3.51);	\$a=5 \$a= - 4
max	max(\$num1,\$num2,...,\$numn)	Tìm số lớn nhất	\$ln=max(1,5,9,4);	\$ln=9
min	min(\$num1,\$num2,...,\$numn)	Tìm số nhỏ nhất	\$nn=min(1,5,9,4);	\$nn=1;
pi	pi()	Lấy giá trị số pi	echo pi();	3.1415926535898
pow	pow(\$var,\$exp)	Trả về giá trị của phép toán \$var mũ \$exp;	echo pow(2,8);	246
rand	rand(\$min,\$max)	Lấy số nguyên ngẫu nhiên có giá trị \$min <rand() <\$max	echo rand(1,10);	Kết quả trên màn hình sẽ là 1 số ngẫu nhiên từ 1 đến 10
round	round(\$val,\$precision)	Làm tròn số \$val với \$precision số lẻ	echo round(1.2434,2);	1.24
sqrt	sqrt(\$var)	Phép toán lấy căn bậc 2 của \$var	echo sqrt(9);	3

### Một số hàm xử lý chuỗi

Tên hàm	Cú pháp	Chức năng	Ví dụ	Kết quả
strlen	strlen(\$str)	Đếm tổng số ký tự có trong \$str	\$str="chao"; echo strlen(\$str);	3

Tên hàm	Cú pháp	Chức năng	Ví dụ	Kết quả
str_word_count	str_word_count(\$str)	Đếm tổng số từ có trong \$str	\$str="mot hai ba"; echo str_word_count(\$str);	3
addslashes	addslashes(\$str)	Thêm vào ký tự \ (back slash) trước các ký tự ' và " trong một chuỗi, có tác dụng tránh lỗi SQL injection khi thêm dữ liệu vào cơ sở dữ liệu	echo addslashes('chao ban "moi"');	chao ban \"moi\"
stripslashes	stripslashes(\$str)	Loại bỏ các ký tự \ trong chuỗi.	\$str='chao ban \\"moi\\"'; stripslashes(\$str);	chao ban “moi”
str_repeat	str_repeat(\$str,\$n)	Lặp lại chuỗi \$str theo \$n lần	echo str_repeat('chao',2) ;	Chaoch ao
str_replace	str_replace(\$chuoi_tim,\$chuoi_thay_the,\$chuoi_nguon)	Tìm kiếm và thay thế chuỗi	\$str="chao ban"; \$str=str_replace('ba n','Tam',\$str); echo \$str;	chao Tam

Tên hàm	Cú pháp	Chức năng	Ví dụ	Kết quả
explode	explode('chuỗi tách',\$str)	Tách chuỗi thành mảng bởi chuỗi tách	\$str='xin chao'; \$array=explode(' ', \$str); echo \$array[1];	chao
implode	Implode('chuỗi nối',\$mang)	Nối các phần tử mảng để tạo thành chuỗi	\$arr=array('xin','chao','ban'); \$str=implode('-', \$arr); echo \$str;	xin-chao-ban
md5	md5(\$str)	Mã hoá chuỗi dạng md5	echo md5('chao');	26729bae1fc5b60a2973324ef387fe5f

#### **Ngoài ra còn một số hàm sinh viên cần tự tham khảo**

substr	Tách chuỗi con từ một chuỗi
strops	Tìm vị trí xuất hiện đầu tiên của chuỗi cần tìm trong chuỗi gốc
strstr	Tách ra một chuỗi con từ vị trí đầu tiên của chuỗi cho trước cho đến cuối chuỗi
strtolower	Chuyển tất cả các ký tự sang chữ thường
strtoupper	Chuyển tất cả các ký tự sang chuỗi hoa

## Một số hàm kiểm tra hợp lệ

ID	Name	Options	Flags	Description
FILTER_VALIDATE_BOOLEAN	"boolean"		FILTER_NULL_ON_FAILURE	Trả về TRUE cho dữ liệu kiểm tra mang giá trị "1", "true", "on" và "yes". Trả về FALSE trong các trường hợp khác. Nếu cò hiệu FILTER_NULL_ON_FAILURE được thiết lập, FALSE sẽ được trả về cho các giá trị "0", "false", "off", "no", và "", Trả về NULL cho tất cả các giá trị không phải là giá trị luận lý (boolean).
FILTER_VALIDATE_EMAIL	"validate_email"			Kiểm tra email có hợp lệ hay không
FILTER_VALIDATE_FLOAT	"float"	Decimal	FILTER_FLAG_ALLOW_THOUSAND	Kiểm tra có phải kiểu số thực không.
FILTER_VALIDATE_INT	"int"	min_range, max_range	FILTER_FLAG_ALLOW_OCTAL, FILTER_FLAG_ALLOW_HEX	Kiểm tra có phải là số nguyên không. Có thể dùng options để giới hạn kiểm tra xem có nằm trong vùng min - max hay không
FILTER_VALIDATE_IP	"validate_ip"		FILTER_FLAG_IPV4, FILTER_FLAG_IPV6, FILTER_FLAG_NO_PRIV_RANGE, FILTER_FLAG_NO_RES_RANGE	Kiểm tra có phải là IP hợp lệ hay không. Hỗ trợ kiểm tra IPv4,IPv6 và kiểm tra IP có không nằm trong vùng IP nội bộ không

FILTER_VALIDATE_REGEXP	"validate_regex"	Regexp		Kiểm tra có khớp với 1 biểu thức chính quy (Regular expression) không.
FILTER_VALIDATE_URL	"validate_url"		FILTER_FLAG_PATH_REQUIRED, FILTER_FLAG_QUERY_REQUIRED	Kiểm tra có phải 1 URL hợp lệ không (Theo chuẩn ở <a href="http://www.faqs.org/rfcs/rfc2396">http://www.faqs.org/rfcs/rfc2396</a> ), Lưu ý là chỉ kiểm tra hợp lệ khi URL chứa các ký tự ASCII. Bất kỳ ký tự unicode nào có trong URL sẽ dẫn đến việc trả về false

**❖ Câu hỏi (bài tập) củng cố:**

**Phản mang:** Cho mảng {1,4,6,2,9,3,7}

1. Viết hàm sắp xếp mảng tăng dần
2. Viết hàm sắp xếp mảng giảm dần.
3. Chèn giá trị cho trước vào vị trí thứ k của mảng với  $0 < k <$  số lượng phần tử của mảng.
4. Tạo mảng cha chứa mảng con như mô tả sau:

Có 3 lớp học, lớp thứ nhất có 2 bạn: Nhân, Đức; lớp thứ 2 có 3 bạn: Lê, Nghĩa, Hiếu; lớp thứ 3 có 2 bạn: Tù, Tôn.

Chú ý: Từ khóa lớp học trong mảng cha được khai báo là: Lớp i; với  $i = 1$  đến 3; (Vì có 3 lớp)

5. Hiển thị Tên của các học sinh theo từ lớp học.
6. Sắp xếp tên của các bạn ở Lớp 2 theo thứ tự các chữ cái giảm dần.

**Phản hàm:**

1. Viết hàm kiểm tra 1 số có phải là số nguyên tố hay không?
2. Viết hàm kiểm tra 1 số là số chẵn hay số lẻ
3. Viết hàm tính tổng hai số
4. Viết hàm tính tổng các số nhỏ hơn n (với n là số nguyên dương)
5. Viết hàm tính tổng các số nguyên tố nhỏ hơn n (với n là số nguyên dương)
6. Viết hàm tính tổng các số chẵn nhỏ hơn bằng n

**Phản tập tin:**

Tạo tập tin baitap.txt rỗng để làm các yêu cầu sau:

1. Viết chương trình nhập chuỗi s="nguyen van a" xuống tập tin baitap.txt
2. Viết chương trình đọc dữ liệu từ tập tin baitap.txt
3. Xây dựng bộ đếm cho website, mỗi khi khách hàng truy cập thì sẽ tự động tăng lên 1 giá trị.

## BÀI 4:

### XỬ LÝ GIÁ TRỊ FORM TRONG PHP

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Lấy và xử lý giá trị trong Form
- Thiết kế form

Một trong những ứng dụng quan trọng của PHP đó là giúp tương tác xử lý dữ liệu trên form của người sử dụng. Nhằm mục đích giúp cập nhật thông tin một cách linh động và dễ dàng quản lý chung hơn bởi sự kết hợp tuyệt vời của cơ sở dữ liệu. Tuy nhiên để làm được điều ấy PHP yêu cầu form phải đáp ứng 1 số quy định chung đặt ra.

Chúng ta cùng phân tích thẻ form trong **HTML** sau:

```
1 <form name="Tên form" action="link xử lý" method="Phương thức">
2
3
```

**Hình 5-4-1: Thẻ <form>**

Chúng ta thấy rằng 1 form phải bao gồm:

**Name:** Tên form để dễ dàng tách biệt với giá trị của chúng.

**Action:** hành động chuyển tiếp đến link xử lý.

**Method:** Là phương thức truyền bao gồm POST và GET.

Ví dụ 5-4-1:

```
1 <form action = "check.php" method = post name=reg>
2 Please type your name here : <BR>
3 <input type = text name = "username"><BR><BR>
4 <input type = submit value = "Submit Data" name="add">
5 </form>
6
```

**Hình 5-4-2: Ví dụ 5-4-1**

Như vậy ta thấy rằng. Đoạn code trên làm những việc sau.

Đầu tiên khi khách nhập liệu username vào thì chúng sẽ chuyển tới trang check.php

để tiến hành xử lý thông tin. Trên phương thức POST, với tên form là reg. Giá trị mà chúng ta gởi là username vừa nhập liệu.

Vậy làm cách nào để chúng ta lấy được giá trị vừa nhập liệu nào ?

PHP cho phép ta lấy giá trị dựa vào 2 phương thức POST và GET.

Đối với **POST** ta có : **`$_POST['Giá trị']`**

Đối với **GET** ta có : **`$_GET['Giá trị']`**

Vậy với đoạn code trên có thể lấy được biến xử lý là : `$_POST['username']`;

Username là tên của field mà người sử dụng nhập liệu vào.

#### 5.4.1. Phương thức GET

Dữ liệu gửi từ trình duyệt lên qua phương thức GET là phần dữ liệu được nhập trực tiếp theo sau địa chỉ URL do trình duyệt gửi lên, được phân biệt với tên file script bằng dấu hỏi chấm (?).

Khi trình duyệt gửi các thông tin này lên máy chủ, PHP sẽ tự động sinh ra một mảng có tên là `$_GET[]` để nắm giữ tất cả các cặp biến và giá trị đó, trong đó, chỉ số của mảng chính là một chuỗi mang tên của tên biến và giá trị của chỉ số đó chính là giá trị của biến do trình duyệt gửi lên.

Ví dụ:

Với url sau: `shownews.php?id=50`

Vậy với trang shownews ta dùng `$_GET['id']` sẽ được giá trị là 50.

#### 5.4.2. Phương thức POST

Đối với những form sử dụng phương thức post (`method='post'`) thì dữ liệu trên form sẽ không hiển thị trên ô địa chỉ. Do đó để lấy giá trị trên form ta sử dụng `$_POST`.

Ví dụ 5-4-2: Xây dựng 1 trang HTML với nội dung gồm form nhập liệu username. Sau đó dùng 1 file php để xuất ra thông tin username mà người sử dụng vừa nhập liệu.

Các bước thực hiện:

Tạo file **userform.html** với nội dung sau:

```
<html>
<form action='processform.php' method='post'>
Nhập tên vào:<br>
<input type='text' name='username'><br>
<input type='submit' value='Hiển thị' name='sbhienthi'>
</form>
</html>
```

Tạo file **processform.php** để xuất ra dữ liệu

```
<?php echo $_POST['username']; ?>
```

**Ghi chú:** Ta có thể sử dụng `$_REQUEST` thay cho `$_GET` và `$_POST`

#### ❖ Câu hỏi (bài tập) cung cấp:

1. Tạo 1 trang web với hộp thoại nhập liệu username và password. Nếu người sử dụng nhập thông tin username/password là admin/12345 thì xuất ra thông báo "welcome, admin" với kiểu chữ Tahoma, màu đỏ. Ngược lại nếu nhập sai thì xuất thông báo "Username hoặc password sai. Vui lòng nhập lại".
2. Bạn hãy viết một chương trình yêu cầu người dùng lựa chọn tính diện tích của: đường tròn, tam giác, chữ nhật. Với lựa chọn là các radio button. Sau đó vận dụng PHP để tính diện tích cho các hình ứng với phần lựa chọn của người dùng
3. Hãy viết chương trình cho phép người dùng nhập vào 1 số. Nếu số đã nhập nhỏ hơn hoặc bằng 12 thì cho biết tháng đó có bao nhiêu ngày. Ngược lại nếu số nhập vào lớn hơn 12 thì trả về kết quả rằng không có tháng này.
4. Hãy viết chương trình xuất ra bảng cửu chương từ 2 => 10
5. Hãy viết chương trình cho phép người dùng nhập vào 1 số nguyên dương  $\leq 10$ , Sau đó trả về kết quả bảng cửu chương của số vừa nhập.
6. Viết chương trình cho phép người dùng nhập vào 1 số, hãy xác định số vừa nhập có phải là số nguyên tố không?

## BÀI 5:

### **KHÁI NIỆM CƠ BẢN VỀ COOKIE VÀ SESSION TRONG PHP**

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Biết cách khai báo và sử dụng biến Cookie
- Biết cách khai báo và sử dụng biến Session

Cookie và session là hai phương pháp sử dụng để quản lý các phiên làm việc giữa người sử dụng và hệ thống. Việc quản lý phiên làm việc này sẽ giúp bạn tạo ra sự chứng thực hiệu quả bởi việc xác nhận thông tin trước khi truy cập vào một phân vùng cố định. Ngoài ra, việc quản lý tốt phiên làm việc cũng giúp người truy cập cảm thấy dễ dàng sử dụng dịch vụ của trang web cho những lần truy cập sau. Bởi có chế quản lý phiên làm việc ghi nhận lại quá trình truy cập của người sử dụng khi họ thăm viéc trang web của bạn lần đầu.

#### **5.5.1. Tổng quan về cookie**

Cookie là 1 đoạn dữ liệu được ghi vào đĩa cứng hoặc bộ nhớ của máy người sử dụng. Nó được trình duyệt gửi ngược lên lại server mỗi khi browser tải 1 trang web từ server. Những thông tin được lưu trữ trong cookie hoàn toàn phụ thuộc vào website trên server. Mỗi website có thể lưu trữ những thông tin khác nhau trong cookie, ví dụ thời điểm lần cuối ta ghé thăm website, đánh dấu ta đã login hay chưa, v.v...

Cookie được tạo ra bởi website và gửi tới browser, do vậy 2 website khác nhau (cho dù cùng host trên 1 server) sẽ có 2 cookie khác nhau gửi tới browser. Ngoài ra, mỗi browser quản lý và lưu trữ cookie theo cách riêng của mình, cho nên 2 browser cùng truy cập vào 1 website sẽ nhận được 2 cookie khác nhau.

Các cookie được sử dụng trong các dịch vụ thương mại điện tử để hỗ trợ chức năng mua hàng trực tuyến, máy chủ có thể theo dõi khách hàng và sao lưu các giao dịch của họ khi họ di chuyển trong site

Mặc dù bản thân cookie thường trú rất hữu ích và thuận tiện nhưng kèm theo đó là những rủi ro có liên quan được lưu trữ.

Cookie ảnh hưởng tới sự riêng tư của người dùng. cũng như rò rỉ thông tin cá nhân. Cookie theo dõi người dùng đã ghé thăm những nơi nào và đã xem những gì trên web. Bản

thân các cookie không thể dùng để phát tán virus, mã độc. Tuy nhiên nó có thể thu thập khá nhiều thông tin cá nhân của bạn nhất là những thông tin bạn cung cấp trên trang web như thông tin thẻ tín dụng... nên cookie có thể làm tăng nguy cơ mất thông tin đăng nhập nếu như người khác sử dụng máy tính của bạn, hoặc trường hợp máy tính của bạn bị xâm nhập, đánh cắp.

#### **5.5.1.1. Thiết lập cookie**

Để thiết lập cookie ta sử dụng cú pháp: Setcookie("tên cookie","giá trị", thời gian sống)

Tên cookie là tên mà chúng ta đặt cho phiên làm việc.

Giá trị là thông số của tên cookie.

Ví dụ:

```
Setcookie ("username", "admin", time () +3600)
```

Như ví dụ trên ta thấy với tên là username và giá trị là admin, có thời gian sống là 1 giờ tính từ thời điểm thiết lập.

Chú ý: Kịch bản cookie phải đặt trên mọi giá trị trả về bao gồm thẻ HTML và lệnh echo.

Kịch bản cookie phải đặt trên mọi giá trị trả về bao gồm thẻ HTML và lệnh echo.

#### **5.5.1.2. Sử dụng cookie**

Để sử dụng lại cookie vừa thiết lập, chúng ta sử dụng cú pháp:

```
Cú pháp: $_COOKIE ["tên cookies"]
```

Tên cookie là tên mà chúng ta thiết lập phía trên.

Ví dụ 5-5-1:

Tạo ra 2 trang (**cookie.php** dùng để khởi tạo cookie có tên là name, nội dung là Nguyen Van A và có thời gian sống là 1 giờ tính từ thời điểm thiết lập) và (**cookie2.php** dùng để hiển thị nội dung của cookie có tên là name đã được khởi tạo ở trang **cookie.php**)

Cách thực hiện:

Nội dung trang **cookie.php**:

```

<?php
    setcookie("name", "Nguyen Van A", time() + 3600);
?>
<html>
<head>
<title>Test page 1</title></head>
<body>
<b><a href=cookie2.php>Click here</a></b>
</body>
</html>

```

Tiếp tục tạo trang **cookie2.php** với nội dung sau:

```

<html>
<head><title>Result Page</title></head>
<body>
<?php
    echo "Ten cua ban la <b>". $_COOKIE['name'] . "</b>";
?>
</body>
</html>

```

### **5.5.1.3. Hủy Cookie**

Để hủy 1 cookie đã được tạo ta có thể dùng 1 trong 2 cách sau:

Cách 1: Gọi hàm setcookie với chỉ duy nhất tên cookie mà thôi.

Cú pháp: `setcookie ("Tên cookie")`

Cách 2: Dùng thời gian hết hạn cookie là thời điểm trong quá khứ.

Ví dụ: `setcookie ("name", "Kenny Huy", time() - 3600);`

## **5.5.2. Tổng quan về session**

Một cách khác quản lý người sử dụng là session. Session được hiểu là khoảng thời gian người sử dụng giao tiếp với 1 ứng dụng. Một session được bắt đầu khi người sử dụng truy cập vào ứng dụng lần đầu tiên, và kết thúc khi người sử dụng thoát khỏi ứng dụng. Mỗi session sẽ có được cấp một định danh (ID) khác nhau và nội dung được lưu trong thư mục thiết lập trong file php.ini (tham số session.save\_path).

### **5.5.2.1. Thiết lập session**

Để thiết lập 1 session ta sử dụng cú pháp: `session_start()`. Đoạn code này phải được nằm trên các kịch bản HTML hoặc những lệnh echo, printf.

Để thiết lập 1 giá trị session, ngoài việc cho phép bắt đầu thực thi session. Chúng ta

còn phải đăng ký 1 giá trị session. Để tiện cho việc gán giá trị cho session đó.

Ta có cú pháp sau: `session_register("Name")`

Ví dụ 5-5-2:

```
<?php  
    session_start();  
    session_register("username");  
?>
```

### 5.5.2.2. Sử dụng giá trị của session

Để sử dụng giá trị của session ta sử dụng mã lệnh sau:

Cú pháp: `$_SESSION["name"]`

Với Name là tên mà chúng ta sử dụng hàm `session_register("name")` để khai báo.

Ví dụ 5-5-3: Tạo trang **session.php** với nội dung sau:

```
<?php  
    session_start();  
    session_register("name");  
    $_SESSION["name"] = "TEST NAME";  
?>  
<html>  
<head>  
<title>Test page 1</title></head>  
<body>  
<b><a href=session2.php>Click here</a></b>  
</body>  
</html>
```

Ví dụ 5-5-4: Tạo trang **session2.php** với nội dung sau:

```
<?php  
    session_start();  
?>  
<html>  
<head><title>Result Page</title></head>  
<body>  
<?php  
    echo "Ten cua ban la <b>". $_SESSION["name"] . "</b>";  
?>  
</body>  
</html>
```

### 5.5.2.3. Hủy bỏ session

Để hủy bỏ giá trị của session ta có thể huỷ bỏ từng session đã đăng ký hoặc huỷ bỏ

toàn bộ các session đã đăng ký.

Để huỷ bỏ tất cả các session đã đăng ký ta sử dụng lệnh: session\_destroy()

Huỷ bỏ 1 session đã đăng ký ta sử dụng lệnh: session\_unset('tên\_session')

Ví dụ 5-5-5: Tạo trang **session3.php** với nội dung sau:

```
<?php
    session_start();
    session_destroy();
?>
<html>
<head>
<title>Test page 1</title></head>
<body>
<b><a href=session2.php>Click here</a></b>
</body>
</html>
```

#### Câu hỏi (bài tập) củng cố:

Xây dựng website sử dụng tính năng session thỏa mãn những yêu cầu sau:

1- Tạo trang đăng nhập với username/password là : admin/test1. Nếu user đăng nhập thành công thì sẽ xuất ra dòng welcome, admin.

2- Tạo trang đăng xuất thoát khỏi phần quản trị

# CHƯƠNG 6

## PHP & MYSQL

### BÀI 1

#### **THAO TÁC TRÊN CƠ SỞ DỮ LIỆU MYSQL**

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Tạo cơ sở dữ liệu (database)
- Tạo bảng (table)
- Thực hiện các thao tác cơ bản (thêm, xóa, sửa, hiển thị trong mysql)

#### **6.1.1. Giới thiệu**

Cơ sở dữ liệu MySQL là tập hợp các đối tượng: bảng, bảng ảo.. cho phép người dùng lưu trữ và truy xuất các thông tin đã được tổ chức và lưu trữ bên trong đó.

#### **6.1.2. Đặc điểm**

Sử dụng cho các ứng dụng Web có quy mô vừa và nhỏ. Để thực hiện các thao tác trên cơ sở dữ liệu MySQL, có thể sử dụng giao diện đồ họa hay dùng dòng lệnh (command line). Các tập tin vật lý lưu trữ cơ sở dữ liệu. Mỗi bảng sẽ được lưu trữ dưới ba tập tin vật lý:

.frm: lưu định dạng (cấu trúc) của bảng

.MYD: lưu nội dung của bảng

.MYI: lưu chỉ mục của bảng

Các tập tin này sẽ được tự động lưu trữ trong thư mục: wamp\mysql\data\tên\_CSDL  
Chiều dài của tên CSDL, bảng, chỉ mục, cột, định danh:

Loại	Chiều dài tối đa (bytes)	Chiều dài tối đa (ký tự không dấu)
CSDL (database)	64	64
Bảng (Table)	64	64
Chỉ mục (Index)	64	64
Cột (Column)	64	64
Định danh (Alias)	255	255

### **Quy tắc đặt tên:**

- Tên không kết thúc bằng khoảng trắng
- Tên cơ sở dữ liệu không có các ký tự: /, \, ., :, \*, ‘’, <, >
- Tên bảng không có các ký tự: /, \, ., :, \*, ‘’, <, >, |
- Chiều dài của tên tối đa là 64 ký tự không dấu. Khi sử dụng các ký tự đa byte thì chiều dài sẽ dựa trên tổng số byte của tất cả các ký tự được dùng.

### **Các thuộc tính của cơ sở dữ liệu:**

- Tên cơ sở dữ liệu: phải duy nhất trong một hệ quản trị cơ sở dữ liệu MySQL
- Vị trí lưu trữ: Khi tạo mới một CSDL hệ thống sẽ tự động tạo ra một thư mục có tên của CSDL và được lưu tại thư mục wamp\mysql\data\

#### ***6.1.2.1. Tạo Cơ sở dữ liệu***

Có hai cách để tạo một Cơ sở dữ liệu: dùng giao diện đồ họa hoặc dùng dòng lệnh

##### **Tạo Cơ sở dữ liệu sử dụng giao diện đồ họa**

Bước 1: khởi động PHPMyAdmin

Bước 2: nhập tên cơ sở dữ liệu muốn tạo vào mục Create New Database và chọn các thông tin khác

Bước 3: Nhấn “Create” để hoàn thành việc tạo cơ sở dữ liệu

##### **Tạo Cơ sở dữ liệu sử dụng câu lệnh CREATE DATABASE**

Cú pháp:

```
CREATE DATABASE Tên_CSDL  
[ [ DEFAULT ] CHARACTER SET <character set name>]  
[ [ DEFAULT ] COLLATE <collation name>]
```

Với:

CHARACTER SET: xác định bộ ký tự mặc định cho CSDL mới

COLLATE: xác định bộ collation

Character set name: tên của một bộ mã bao gồm các ký tự, ký số, và biểu tượng để

lưu trữ các thông tin trong CSDL

Collation name: tên của bộ mã tùy theo từng khu vực dựa trên bộ mã chuẩn character set name

#### Ví dụ 6-1-1: **Tạo CSDL ql\_ban\_sua**

```
CREATE DATABASE ql_ban_sua CHARACTER SET utf-8 COLLATE  
utf8_unicode_ci
```

Hay

```
CREATE DATABASE ql_ban_sua
```

#### **6.1.2.2. Xóa Cơ sở dữ liệu**

Có hai cách xóa cơ sở dữ liệu: Dùng giao diện đồ họa hoặc dùng dòng lệnh.

##### **a. Giao diện đồ họa**

Bước 1: Chọn Cơ sở dữ liệu cần xóa, sau đó nhấn DROP

Bước 2: Xác nhận việc xóa Cơ sở dữ liệu đã chọn, nhấn OK để xóa

##### **b. Dùng câu lệnh *DROP DATABASE***

Cú pháp:

```
DROP DATABASE TEN_CSDL
```

#### Ví dụ 6-1-2: Xóa CSDL ql\_ban\_sua

```
DROP DATABASE ql_ban_sua
```

#### **6.1.2.3. Thao tác với bảng trong MySQL**

Dùng để lưu trữ thông tin của những đối tượng, thực thể trong thế giới thực muốn được lưu trữ vào máy tính.

Các thông tin trong bảng sẽ được tổ chức thành các dòng (row) và các cột (column).

Mỗi dòng thông tin trong bảng là duy nhất do có một hoặc nhiều cột làm khoá chính. Dữ liệu của cột làm khoá chính không trùng lặp trong bảng.

Các bảng thường có quan hệ với nhau giúp trao đổi và chia sẻ thông tin. Tên bảng là

do người dùng tạo ra và duy nhất trong cơ sở dữ liệu.

#### **Các thuộc tính của cột trong bảng:**

Tên cột là do người dùng đặt ra và duy nhất trong bảng. Ví dụ: đặt tên cho cột mã sữa trong bảng sữa là: Ma\_Sua.

Kiểu dữ liệu : xác định loại dữ liệu được lưu trữ trên từng cột.

Các kiểu dữ liệu số nguyên:

Kiểu dữ liệu	Kích thước	Miền giá trị
Tinyint	1 byte	-127 => 128 hay 0..255
Smallint	2 bytes	-32768 => 32767 hay 0..65535
Mediumint	3 bytes	-8388608 => 838860 hay 0..16777215
Int	4 bytes	- $2^{31}$ => $2^{31}-1$ hay 0.. $2^{32}-1$
Bigint	8 bytes	- $2^{63}$ => $2^{63}-1$ hay 0.. $2^{64}-1$

Các kiểu dữ liệu TRUE/FALSE

Kiểu dữ liệu	Kích thước	Miền giá trị
Bool / boolean	1 byte	Có hai giá trị là True và False

Các kiểu dữ liệu số thực

Kiểu dữ liệu	Kích thước	Miền giá trị
float	4 bytes	-3.402823466E+38 => -1.175494351E-38; 0; 1.175494351E-38 => 3.402823466E+38
Double	8 bytes	-17976931348623157E+308=>-2.2250738585072014E-308;0; 2.2250738585072014E-308=> 1.7976931348623157E+308

### Các kiểu dữ liệu

Kiểu dữ liệu	Kích thước	Miền giá trị
Date	'1000-01-01' => '9999-12-31'	Ngày với định dạng yyyy-mm-dd
Datetime	'1000-01-01 00:00:00' => '9999-12-31 23:59:59'	Ngày giờ với định dạng yyyy-mm-dd hh:mm:ss
Year[(2 4)]	4 ký số: '1901'=>'2155' 2 ký số: '1970'=>'2069'	Năm với định dạng 2 ký số hoặc 4 ký số
Timestamp[(kích cỡ định dạng)]	'1970-01-01 00:00:01'	Timestamp trình bày dưới dạng yyyy-mm-dd hh:mm:ss
Time	'00:00:00' => '23:59:59'	Giờ với định dạng hh:mm:ss

### Các kiểu dữ liệu chuỗi

Kiểu dữ liệu	Kích thước	Miền giá trị
Char	1=>255 ký tự	Chuỗi cố định
Varchar	1=>255 ký tự	Chuỗi động
Tinyblob	1=> $2^8$ -1 bytes (255 bytes)	Kiểu đối tượng nhị phân cỡ 255 ký tự
Tinytext	1=> $2^8$ -1 bytes (255 ký tự)	Kiểu đối tượng chuỗi kích cỡ 255 ký tự
Blob	1=> $2^{16}$ -1 bytes (65535 bytes)	Kiểu blob cỡ 65535 ký tự
Text	1=> $2^{16}$ -1 ký tự (65535 ký tự)	Kiểu chuỗi dạng văn bản cỡ 65535 ký tự

Mediumblob	$1 \Rightarrow 2^{24} - 1$ bytes(16777214 bytes)	Kiểu blob vừa cỡ 16777215 ký tự
------------	--	---------------------------------

Độ dài dữ liệu (length / value): Quy định độ dài dữ liệu mà cột sẽ lưu trữ đối với kiểu dữ liệu chuỗi hoặc số

Kiểu hiển thị (collation): Quy định bảng mã hiển thị cho dữ liệu cột. Ví dụ: utf8\_unicode\_ci

Thuộc tính (attribute): Quy định thuộc tính cho cột, mặc định là không quy định.

Cho phép để trống dữ liệu (NULL): Quy định cột có thể để trống hay không khi thêm, cập nhật dữ liệu.

Giá trị mặc định (default): là giá trị sẽ thêm vào cho cột khi thêm mới mẫu tin mà người dùng không nhập giá trị cho cột này.

Thuộc tính mở rộng (extra): cho phép thiết lập thuộc tính auto\_increment (cột có giá trị tự động tăng dần khi thêm mới mẫu tin) cho khoá chính.

Ghi chú (comment): chuỗi chú thích cho cột.

#### a. *Tạo bảng*

##### Cách tạo bảng đơn giản

Cú pháp:

```
Create table ten_bang
(
    Ten_cot_1 kieu_du_lieu[ (kich_co) ] [Not NULL],
    Ten_cot_2 kieu_du_lieu[ (kich_co) ] [Not NULL],
    ...
)
```

Chú ý: NOT NULL có nghĩa là không cho phép để trống dữ liệu trong cột.

Ví dụ 6-1-3: tạo bảng hang\_sua với hai cột dữ liệu không được phép bỏ trống là Ma\_hang\_sua và Ten\_hang\_sua

```
CREATE TABLE hang_sua (
    Ma_hang_sua varchar(20) NOT NULL,
    Ten_hang_sua varchar(100) NOT NULL,
    Dia_chi varchar(200),
    Dien_thoai varchar(20),
    Email varchar(100)
)
```

#### Cách tạo bảng thiết lập khoá chính (primary key)

Cú pháp: primary key (cac\_cot\_tham\_gia\_lam\_khoa\_chinh).

Chú ý: Đối với bảng mà khoá chính chỉ có một cột thì có thể khai báo khoá chính ngay sau khi khai báo cột. Đối với bảng có nhiều cột tham gia làm khoá thì thiết lập các cột trước rồi mới tạo khoá chính sau.

Ví dụ 6-1-4: tạo bảng hang\_sua như trên với khoá chính là cột Ma\_hang\_sua

```
CREATE TABLE hang_sua (
    Ma_hang_sua varchar(20) NOT NULL PRIMARY
    KEY,
    Ten_hang_sua varchar(100) NOT NULL,
    Dia_chi varchar(200),
    Dien_thoai varchar(20),
    Email varchar(100)
)
```

#### **b. Thay đổi cấu trúc bảng**

Có thể thay đổi lại cấu trúc bảng mà không làm mất đi dữ liệu đã có trong bảng bằng cách sử dụng câu lệnh ALTER TABLE.

Thêm cột mới trong bảng

Cú pháp:

```
ALTER TABLE Ten_bang
ADD
Ten_cot kieu_du_lieu[ (kich_co) ] [...]
```

Chú ý: tên cột mới thêm vào phải khác với tên cột đã có trong bảng.

Sửa đổi kiểu dữ liệu của cột

Cú pháp:

```
ALTER TABLE ten_bang  
    CHANGE ten_cot_cu ten_cot_moi kieu_du_lieu_moi [kich_co]
```

Chú ý: câu lệnh này có thể dùng để thay đổi tên của cột thành tên mới.

#### c. *Huỷ cột trong bảng*

Khi không cần sử dụng cột trong bảng, chúng ta có thể sử dụng cú pháp ALTER TABLE để huỷ bỏ cột.

Cú pháp:

```
ALTER TABLE Ten_bang  
    DROP COLUMN Ten_cot,..
```

Ví dụ 6-1-5: Huỷ bỏ cột Email trong bảng hang\_sua đã tạo ở trên.

```
ALTER TABLE hang_sua  
    DROP COLUMN email
```

#### d. *Xoá bảng*

Để xoá bảng trong cơ sở dữ liệu, ta sử dụng lệnh DROP TABLE.

Cú pháp:

```
DROP TABLE Ten_bang
```

Ví dụ 6-1-6:

```
DROP TABLE hang_sua
```

#### 6.1.2.4. *Import và export dữ liệu*

Trong quá trình làm việc với cơ sở dữ liệu, việc sao lưu và phục hồi (import / export) dữ liệu là hết sức quan trọng. Chúng ta có thể di chuyển dữ liệu từ máy client lên máy server hoặc người lại nhờ vào hai bước này. Export là để sao lưu dữ liệu lại còn Import là dùng để phục hồi dữ liệu.

a. *Import dữ liệu*: Nhập dữ liệu từ bên ngoài vào Database trong MySQL

Bước 1: Vào Database muốn import dữ liệu

Bước 2: Chọn chức năng import

Bước 3: Chọn tập tin chứa dữ liệu cần import ở ô *Location of the text file*

Bước 4: Xác nhận (nhấn Go) để hoàn thành việc import dữ liệu

b. **Export dữ liệu:** Xuất dữ liệu từ Database trong MySQL ra tập tin có dạng \*.sql:

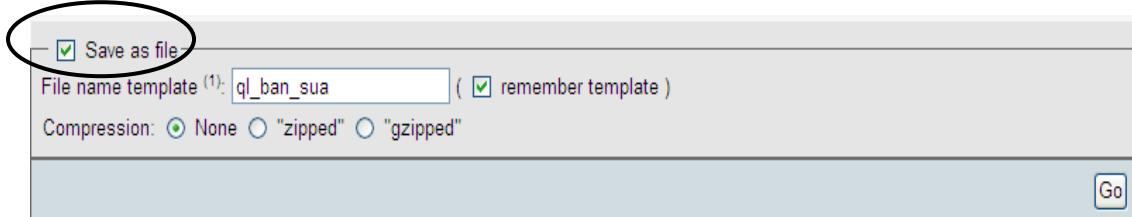
Bước 1: Chọn Database muốn export dữ liệu

Bước 2: Chọn chức năng export

Bước 3: Chọn các bảng trong Database cần export

Bước 4: Chọn loại file để export

Bước 5: Thiết lập các thuộc tính cho file export, chọn *Save as file*



**Hình 6-1-1: Export dữ liệu**

Bước 6: Đặt tên cho file export (nhấn Go để qua phần chọn nơi lưu trữ)

Bước 7: Chọn nơi lưu trữ file export

Bước 8: Click “Save” để hoàn thành việc export dữ liệu

❖ Câu hỏi (bài tập) củng cố:

Thiết kế CSDL tên là QLBanHang

Thực hiện các thao tác export và import trong mysql đối với CSDL QLBanHang

## BÀI 2

### **KẾT NỐI CƠ SỞ DỮ LIỆU MYSQL**

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Kết nối PHP với MySQL

Trong bài học này chúng ta sẽ kết nối từ PHP tới cơ sở dữ liệu MySQL.

#### **6.2.1. Các bước xây dựng chương trình có kết nối tới Cơ sở dữ liệu**

Thông thường, trong một ứng dụng có giao tiếp với Cơ sở dữ liệu, ta phải làm theo bốn trình tự sau:

Bước 1: Thiết lập kết nối tới cơ sở dữ liệu.

Bước 2: Lựa chọn Cơ sở dữ liệu.

Bước 3: Tiến hành các truy vấn SQL, xử lý các kết quả trả về nếu có

Bước 4: Đóng kết nối tới Cơ sở dữ liệu.

Nếu như trong lập trình thông thường trên Windows sử dụng các chương trình điều khiển trung gian (ADO, ODBC...) để thực hiện kết nối và truy vấn, thì trong PHP, khi lập trình tương tác với cơ sở dữ liệu, chúng ta thường sử dụng thông qua các hàm.

**Chú ý:** Trước khi sử dụng PHP để lập trình với MySQL, hãy sử dụng chương trình quản lý phpMyAdmin để tạo trước một cơ sở dữ liệu, vài bảng cũng như người dùng... để tiện thực hành. Bài hướng dẫn sử dụng phpMyAdmin sẽ được trình bày sau.

#### **Thiết lập kết nối tới MySQL**

Cú pháp:

```
mysql_connect(servername, username, password);
```

*servername*: tham số tùy chọn, xác định server cần phải kết nối tới. Giá trị phục thuộc vào nhà cung cấp host. Khi chúng ta thử nghiệm với việc cài đặt wamp hay bất kỳ webserver nào khác thì servername mặc định ở đây là “localhost”.

*username*: tham số tùy chọn, xác định tên người dùng đăng nhập vào hệ thống. Giá

trị mặc định là tên của người dùng làm chủ tiến trình của server.

*password*: tham số tùy chọn, xác định mật khẩu của người dùng.

Ví dụ 6-2-1:

```
<?php  
    mysql_connect ("localhost", "root", "");  
?>
```

Khi kết nối tới MySQL thành công, hàm sẽ trả về giá trị là một số nguyên định danh của liên kết, ngược lại, hàm trả về giá trị false.

Ta có thể dùng hàm if để kiểm tra xem có kết nối được tới MySQL hay không, hãy xem ví dụ 6-2-2:

```
<?php  
    $link = mysql_connect("localhost", "mysql_user",  
                          "mysql_password");  
    if (!$link)  
    {  
        echo "Không thể kết nối được tới CSDL MySQL";  
    }  
?>  
Đoạn code bên trên có thể được viết ngắn gọn như sau:  
<?php  
    $link = mysql_connect("localhost", "mysql_user",  
                          "mysql_password") or die ("Không thể kết nối  
được tới CSDL MySQL");  
?>
```

### Lựa chọn CSDL

Để lựa chọn một CSDL (cơ sở dữ liệu) nào đó mà người sử dụng có tên là *tên\_truy\_cập* có quyền sử dụng, ta dùng hàm *mysql\_select\_db*:

Cú pháp:

```
mysql_select_db (tên_CSDL);
```

Hàm này chỉ được dùng sau khi thiết lập kết nối bằng hàm *mysql\_connect*. Hàm này trả về TRUE nếu thành công, FALSE nếu thất bại.

Ví dụ 6-2-3:

```

<?php
$link = mysql_connect("localhost", "mysql_user",
                      "mysql_password");
if (!$link)
{
    echo "Không thể kết nối được tới MySQL";
}
if (!mysql_select_db ("Tên_CSDL"))
{
    echo "Không thể lựa chọn được CSDL";
}
?>
Đoạn code trên có thể viết lại như sau:
<?php
$link = mysql_connect("localhost", "mysql_user",
                      "mysql_password") or die("Không thể kết nối được
tới MySQL");
mysql_select_db ("Tên_CSDL") or die("Không thể lựa chọn được
CSDL");
?>

```

### **Đóng kết nối tới CSDL:**

Để đóng kết nối tới CSDL, ta dùng hàm mysql\_close.

Hàm này có nhiệm vụ đóng kết nối tới CSDL có mã định danh được tạo ra bởi hàm mysql\_connect().

Cú pháp:

```
mysql_close(connection);
```

connection: tham số tùy chọn, xác định kết nối. Nếu không xác định thì kết nối cuối cùng được mở bởi hàm mysql\_connect() hoặc hàm mysql\_pconnect() sẽ được sử dụng.

Kết quả trả về là TRUE nếu đóng kết nối thành công, ngược lại sẽ trả về FALSE nếu thất bại.

Ví dụ 6-2-4:

```

<?php
$link = mysql_connect("localhost", "mysql_user",
                      "mysql_password") or die("Không thể kết nối được
tới MySQL");
mysql_close($link);
?>

```

### 6.2.2. Thông báo lỗi

Trong quá trình làm việc với CSDL lỗi có thể phát sinh cần thông báo những lỗi phát sinh này bằng cách sử dụng hàm **mysql\_error()**.

Cú pháp:

```
mysql_error(connection);
```

Kết quả trả về là câu thông báo lỗi nếu có lỗi phát sinh, ngược lại kết quả trả về sẽ là một chuỗi rỗng “”.

Kết hợp hàm **mysql\_error()** với hàm **die()** hoặc hàm **exit()** để vừa thông báo lỗi vừa kết thúc công việc

Ví dụ 6-2-5: thông báo lỗi nếu không thể tạo kết nối

```
<?php  
$db = mysql_connect("localhost", "root", "") or die('Không thể  
kết nối: ' . mysql_error());  
?>
```

#### ❖ Câu hỏi (bài tập) củng cố:

1. Kết nối với máy chủ, nếu kết nối thành công thì báo là “Kết nối thành công” ngược lại báo “Kết nối thất bại”
2. Tạo CSDL Demo, Kết nối với CSDL Demo, nếu kết nối thành công thì báo “Kết nối CSDL thành công” ngược lại báo “Không tồn tại CSDL Demo”

## BÀI 3

### **THAO TÁC TRÊN CƠ SỞ DỮ LIỆU MYSQL VỚI PHP**

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Đếm số lượng mẫu tin
- Hiển thị dữ liệu
- Lưu thông tin mới vào CSDL
- Cập nhật dữ liệu
- Xóa dữ liệu

#### **6.3.1. Đếm số lượng mẫu tin**

Cú pháp:

```
mysql_num_rows (data);
```

*data*: là tham số bắt buộc. Xác định con trỏ dữ liệu. Con trỏ dữ liệu là kết quả trả về của hàm mysql\_query().

Hàm này có kết quả trả về là số lượng mẫu tin nếu thành công, ngược lại kết quả trả về sẽ là FALSE nếu thất bại.

Ví dụ 6-3-1: đếm số lượng mẫu tin có trong bảng hang\_sua

```
$sql= "SELECT * FROM hang_sua";
$result = mysql_query($sql);
echo mysql_num_rows($result);
```

#### **6.3.2. Hiển thị dữ liệu**

Sử dụng một trong các cách sau:

Duyệt dữ liệu theo dạng mỗi mẫu tin là một dòng

Duyệt theo dạng mỗi mẫu tin là một mảng

Duyệt theo dạng mỗi mẫu tin là một đối tượng

**Duyệt dữ liệu theo dạng mỗi mẫu tin là một dòng mysql\_fetch\_row()**

Cú pháp:

```
mysql_fetch_row(data);
```

*data*: là tham số bắt buộc. Xác định con trỏ dữ liệu. Con trỏ dữ liệu là kết quả trả về của hàm mysql\_query()

Kết quả trả về một mảng (có chỉ số) chứa giá trị của một dòng dữ liệu với mỗi phần tử là nội dung của một cột → truy cập bằng cách gọi từng phần tử của mảng \$row[0], \$row[1], \$row[2],...

Duyệt dữ liệu theo dạng mỗi mẫu tin là một dòng mysql\_fetch\_row()

Ví dụ 6-3-2:

```
mysql_select_db("ql_ban_sua");
$result = mysql_query("SELECT * FROM KHACH_HANG");
if (mysql_num_rows($result)<>0) {
    print_r(mysql_fetch_row($result));
}
mysql_close($db);
```

Duyệt tất cả các mẫu tin: kết hợp cấu trúc lặp while và hàm mysql\_fetch\_row() đặt trong cấu trúc lặp while.

Ví dụ 6-3-3: duyệt và in tất cả các mẫu tin có trong bảng khách hàng

```
if (mysql_num_rows($result)<>0)
{
    while($row = mysql_fetch_row($result)) {
        print_r($row);
    }
}
```

**Duyệt dữ liệu theo dạng mỗi mẫu tin là một mảng mysql\_fetch\_array()**

Cú pháp:

```
mysql_fetch_array(data);
```

*data*: là tham số bắt buộc. Xác định con trỏ dữ liệu. Con trỏ dữ liệu là kết quả trả về của hàm mysql\_query()

Kết quả trả về là một mảng (có chỉ số chuỗi) chứa giá trị của một dòng dữ liệu với mỗi phần tử là nội dung của một cột → truy cập bằng cách gọi từng phần tử của mảng:

`$row[“tên cột 1”], $row[“tên cột 2”], $row[“tên cột 3”],...`

Ví dụ 6-3-4:

```
if (mysql_num_rows ($result)<>0) {  
    echo (mysql_fetch_array ($result));  
}
```

Duyệt tất cả các mẫu tin: kết hợp cấu trúc lặp while và hàm `mysql_fetch_array()` đặt trong cấu trúc lặp while.

Ví dụ 6-3-5:

```
if (mysql_num_rows ($result)<>0) {  
    while ($array = mysql_fetch_array ($result)) {  
        echo ($array);  
    }  
}
```

**Duyệt dữ liệu theo dạng mỗi mẫu tin là một đối tượng `mysql_fetch_object()`**

Cú pháp:

```
mysql_fetch_object (data)
```

`data`: là tham số bắt buộc. Xác định con trỏ dữ liệu. Con trỏ dữ liệu là kết quả trả về của hàm `mysql_query()`

Kết quả trả về là một mẫu tin trong bộ các mẫu tin như là một đối tượng → truy cập bằng cách gọi từng thuộc tính của đối tượng: `$tên_đối_tượng → tên_cột_1, $tên_đối_tượng → tên_cột_2,...`

Ví dụ 6-3-6:

```
if (mysql_num_rows ($result) <>0) {  
    print_r (mysql_fetch_object ($result));  
}
```

Duyệt tất cả các mẫu tin: kết hợp cấu trúc lặp while và hàm `mysql_fetch_object()` đặt trong cấu trúc lặp while.

Ví dụ 6-3-7: duyệt và in tất cả các mẫu tin có trong bảng khách hàng

```
if (mysql_num_rows ($result) <>0) {  
    while ($object = mysql_fetch_object ($result)) {  
        print_r ($object);  
    }  
}
```

### **Lưu ý:**

**Hiển thị dữ liệu không định dạng:** cách đơn giản nhất là hiển thị dữ liệu theo dạng bảng hoặc dạng text.

### **Hiển thị dữ liệu có yêu cầu:**

**Định dạng hiển thị dữ liệu:** Yêu cầu đơn giản nhất của một trang Web là dữ liệu hiển thị phải được định dạng giúp cho người dùng dễ dàng đọc thông tin. Các định dạng thông thường là định dạng cột, dòng, kích thước, màu sắc, kiểu chữ,... cho nội dung hiển thị.

## THÔNG TIN KHÁCH HÀNG

Mã KH	Tên khách hàng	Giới tính	Địa chỉ	Số điện thoại
kh001	Khuất Thùy Phương	1	A21 Nguyễn Oanh quận Gò Vấp	9874125
kh002	Đỗ Lâm Thiên	0	357 Lê Hồng Phong Q.10	8351056
kh003	Phạm Thị Nhung	1	56 Đinh Tiên Hoàng quận 1	9745698
kh004	Nguyễn Khắc Thiện	0	12bis Đường 3-2 quận 10	8769128
kh005	Tô Trần Hồ Giàng	0	75 Nguyễn Kiệm quận Gò Vấp	5792564
kh006	Nguyễn Kiến Thi	1	357 Lê Hồng Phong Q.10	9874125
kh008	Nguyễn Anh Tuấn	0	1/2bis Nơ Trang Long Q.BT TP.HCM	8753159

**Hình 6-3-1: Hiển thị dữ liệu dạng bảng**

Để thực hiện được các yêu cầu về định dạng → cần phải xem xét và tính toán cho các nội dung cần hiển thị.

Ví dụ 6-3-8: hiển thị thông tin khách hàng với định dạng như sau:

Dòng tiêu đề của bảng sẽ được canh giữa, chữ in đậm, màu sắc phân biệt so với các dòng khác.

Các dòng thông tin của khách hàng: dòng chẵn và dòng lẻ có màu nền khác nhau, cột giới tính được canh giữa

### **Tạo các cột tùy biến**

Nhu cầu hiển thị dữ liệu một cách rõ ràng, chi tiết, dễ hiểu trở thành một yêu cầu tất yếu khi thiết kế trang Web.

## THÔNG TIN KHÁCH HÀNG

Mã KH	Tên khách hàng	Giới tính	Địa chỉ	Số điện thoại
kh001	Khuất Thùy Phương	1	A21 Nguyễn Oanh quận Gò Vấp	9874125
kh002	Đỗ Lâm Thiên	0	357 Lê Hồng Phong Q.10	8351056
kh003	Phạm Thị Nhung	1	56 Đinh Tiên Hoàng quận 1	9745698
kh004	Nguyễn Khắc Thiện	0	12bis Đường 3-2 quận 10	8769128
kh005	Tô Trần Hồ Giang	0	75 Nguyễn Kiệm quận Gò Vấp	5792564
kh006	Nguyễn Kiến Thi	1	357 Lê Hồng Phong Q.10	9874125
kh008	Nguyễn Anh Tuấn	0	1/2bis Nơ Trang Long Q.BT TP.HCM	8753159

**Hình 6-3-2: Ví dụ 6-3-8 – Trang thông tin khách hàng**

Một trong những cách hiển thị dữ liệu đáp ứng nhu cầu đó là tạo các cột tùy biến

Ví dụ 6-3-9: trong CSDL thường dùng kiểu dữ liệu boolean cho cột giới tính → cột này sẽ có một trong hai giá trị là 0 hoặc 1 và tùy theo quy ước để xác định 0 hoặc 1 thuộc giới tính gì. Nếu cột giới tính hiển thị 0, 1 và sau đó lại có dòng chú giải thì sẽ không thuận tiện → dùng hình ảnh hoặc checkbox,... để hiển thị giới tính. Cách hiển thị này vừa đẹp mắt, vừa dễ hiểu.

### Phân trang

Đối với những bảng có số lượng mẫu tin lớn khi hiển thị dữ liệu sẽ làm cho người dùng cảm thấy khó xem → thực hiện việc phân trang.

Phân trang dữ liệu giúp cho việc xem và tìm kiếm thông tin dễ dàng, nhanh chóng và thuận tiện.

Kết nối với CSDL và máy chủ

```
<?php  
$kn=mysql_connect("localhost","root","mật_khẩu") or die("lỗi  
kết nối");  
mysql_select_db("ql_ban_sua");
```

Đếm số dòng trong CSDL:

```
$sql="select * from hang_sua";  
$rs=mysql_query($sql);  
$sodong=mysql_num_rows($rs);
```

Cho phép mỗi trang hiển thị tối đa 2 dòng dữ liệu: \$trang=2;

Dựa vào số dòng trong CSDL và số dữ liệu tối đa hiển thị để tìm ra số trang

```

if ($sodong%$strang==0)
{
    $strang=$sodong/$strang;
}
else
{
    $strang= ( ($sodong-($sodong%$strang)) /$strang)+1;
}

```

Hiện thị số trang ra màn hình để cho phép người dùng chọn trang muốn xem

```

for($i=1;$i<=$strang;$i++)
{
    echo "<a href=hienthi.php?n=$i>$i</a> << ";
}

```

Kiểm tra xem dữ liệu truyền trên trình duyệt có hợp lệ không. Nếu biến n trên đường truyền được thiết lập và nếu n là số và n lớn hơn 0 và n nhỏ hơn bằng số trang thì n hợp lệ ngược lại gán giá trị mặc định n=1;

```

if(isset($_REQUEST['n']) && is_numeric($_REQUEST['n']) &&
$_REQUEST['n']>0 && $_REQUEST['n']<=$strang)
    $n=$_REQUEST['n'];
else
    $n=1;

```

Gán dữ liệu bắt đầu lấy trong csdl là biến \$bat\_dau với:

```
$bat_dau=($n-1) * $strang;
```

Sử dụng chức năng limit để hiện thị dữ liệu ứng với từng trang.

```
$sql1="select * from hang_sua limit $bat_dau,$strang";
```

Lưu ý: Limit 0,5 tức là lấy 5 dòng bắt đầu từ 0. limit 5,5 tức là lấy 5 dòng bắt đầu từ dòng 5.

Hiển thị thông tin ra trình duyệt gồm tên khách hàng, địa chỉ và số điện thoại

```

$rs1=mysql_query($sql1);
while($r=mysql_fetch_array($rs1))
{
?>

```

```

<tr><td><?php echo $r[1] ;?></td>
<td><?php echo $r['3'] ;?></td>
<td><?php echo $r['4'] ;?></td></tr>
<?php
}
?>
```

## Liên kết trang có chuỗi tham số

Chuỗi URL có cấu trúc định dạng như sau:

[http://<host>\[:<port>\]\[<path>\]\[?<param>\\_value\]](http://<host>[:<port>][<path>][?<param>_value])

Phần cuối của chuỗi URL là chuỗi tham số có cú pháp như sau:

[?Tham\_số\_1=Giá\_trị\_1[&Tham\_số\_2=Giá\_trị\_2[&...]]]

Chú ý: trong trường hợp có nhiều tham số, các cặp [Tham\_số=Giá\_trị] phân cách nhau bằng dấu &

Ví dụ: Khi người dùng muốn xem thông tin chi tiết của sữa, người dùng click chọn sữa đó, URL sẽ có đường dẫn là địa chỉ của trang chi tiết sữa và chuỗi tham số là mã sữa

	<u>Sữa đặc có đường Ông Thọ Trắng</u>	360	11.000
---	---------------------------------------	-----	--------

Trang danh sách sữa  
[http://localhost/MBOOK\\_PHP/phan\\_3/c4\\_chi\\_tiet\\_sua\\_2.php?ma\\_sua=VNM001](http://localhost/MBOOK_PHP/phan_3/c4_chi_tiet_sua_2.php?ma_sua=VNM001)

## Sữa đặc có đường Ông Thọ Trắng - Vinamilk



### Thành phần dinh dưỡng:

Được chế biến từ sữa bò cao cấp, bột sữa, chất béo, đường kính và vitamin A,D,B1 theo tiêu chuẩn quốc tế về giá trị và vệ sinh thực phẩm

### Lợi ích:

Uống sữa Ông Thọ mỗi ngày sẽ giúp chống suy dinh dưỡng, giải độc, tăng tuổi thọ, phát triển chiều cao, tăng cường sức đề kháng.

**Trọng lượng: 360 gr - Đơn giá: 11.000 VNĐ**

## Nhận giá trị truyền qua tham số:

`$_REQUEST["tên_tham_số"];`

Ví dụ: nhận giá trị của tham số mã sữa

```
$ma_sua = $_REQUEST["ma_sua"];
```

### 6.3.3. Lưu thông tin mới vào CSDL

Sử dụng hàm mysql\_query() kết hợp với câu lệnh truy vấn INSERT INTO.

Hàm này được dùng để gửi một truy vấn (hiển thị thông tin, thêm mới, xoá, cập nhật) tới một kết nối MySQL.

Ví dụ 6-3-10: Thêm mới một hãng sửa với giá trị cụ thể vào bảng hãng sửa đã có.

```
$db = mysql_connect("localhost", "root", "");  
if (!$db)  
{  
    die("Không thể kết nối CSDL: " . mysql_error());  
}  
mysql_select_db("ql_ban_sua", $db);  
mysql_query("INSERT INTO hang_sua VALUES ('XO', 'XO', 'Khu  
công nghiệp Đồng Nai', '7456321', 'xo@xo.com')");  
mysql_close($db);
```

### 6.3.4. Cập nhật dữ liệu

Sử dụng hàm mysql\_query() kết hợp với câu lệnh truy vấn UPDATE.

Hàm này được dùng để gửi một truy vấn (hiển thị thông tin, thêm mới, xoá, cập nhật) tới một kết nối MySQL.

Ví dụ 6-3-11: cập nhật lại địa chỉ cho hãng sửa XO là ‘Khu công nghiệp Bình Dương’

```
$db = mysql_connect("localhost", "root", "");  
if (!$db){  
    die("Không thể kết nối CSDL: " . mysql_error());  
}  
mysql_select_db("ql_ban_sua", $db);  
mysql_query("UPDATE hang_sua SET Dia_chi = 'Khu công nghiệp  
Bình Dương' WHERE Ma_hang_sua = 'XO'");  
mysql_close($db);
```

### 6.3.5. Xóa dữ liệu

Sử dụng hàm mysql\_query() kết hợp với câu lệnh truy vấn DELETE.

Hàm này được dùng để gửi một truy vấn (hiển thị thông tin, thêm mới, xoá, cập nhật) tới một kết nối MySQL

Ví dụ 6-3-12: xoá thông tin của hãng sửa XO

```
$db = mysql_connect("localhost", "root", "");  
if (!$db) {  
    die("Không thể kết nối: " . mysql_error());  
}  
mysql_select_db("ql_ban_sua", $con);  
mysql_query("DELETE FROM hang_sua WHERE Ma_hang_sua='XO'");  
mysql_close($db);
```

Lưu ý: PHP có thể kết hợp với các CSDL khác

PHP có thể thực thi được trên cả hai môi trường là Windows và Linux do đó PHP có thể kết nối nhiều loại CSDL (MS Access, SQL Server, ...) trên hai môi trường này.

Dùng ODBC để kết nối tới nguồn dữ liệu. Với một kết nối ODBC, có thể kết nối đến nhiều CSDL khác nhau

#### ❖ Câu hỏi (bài tập) củng cố:

1. Làm lại tất cả các ví dụ mẫu
2. Tạo Cơ sở dữ liệu sau:

Tên csdl: QLSV

Tạo các bảng:

tblSinhvien(MaSV, TenSV, NamSinh, QueQuan, MaLop)

tblLop(MaLop, TenLop, MaNG)

tblNghanh(MaNG, TenNG)

tblMonHoc(MaMH, TenMH, TCLT, TCTH)

tblSinhVien\_MonHoc(MaSV, MaMH, HK\_NK, Diem)

Giải thích:

Mỗi sinh viên có duy nhất 1 Mã sinh viên (MaSV), được học ở một lớp (tblLop), cần quản lý thêm thông tin về tên sinh viên, năm sinh, quê quán của sinh viên.

Mỗi lớp có duy nhất 1 Mã Lớp (MaLop) xác định tên lớp và thuộc một ngành (tblNghanh).

Một ngành học có duy nhất một Mã Ngành (MaNG), từ mã ngành sẽ xác định được tên ngành.

Một môn học có duy nhất một Mã Môn Học (MaMH), cần quản lý thêm thông tin cụ thể tên môn, số tín chỉ lý thuyết và số tín chỉ thực hành.

Một sinh viên khi học xong một môn học vào một học kỳ nào đó sẽ được cập nhật lại điểm số.

3. Dùng phpMyAdmin thực hiện thêm, xóa, sửa, tìm kiếm, sắp xếp, hiển thị thông tin sinh viên.

## CHƯƠNG 7

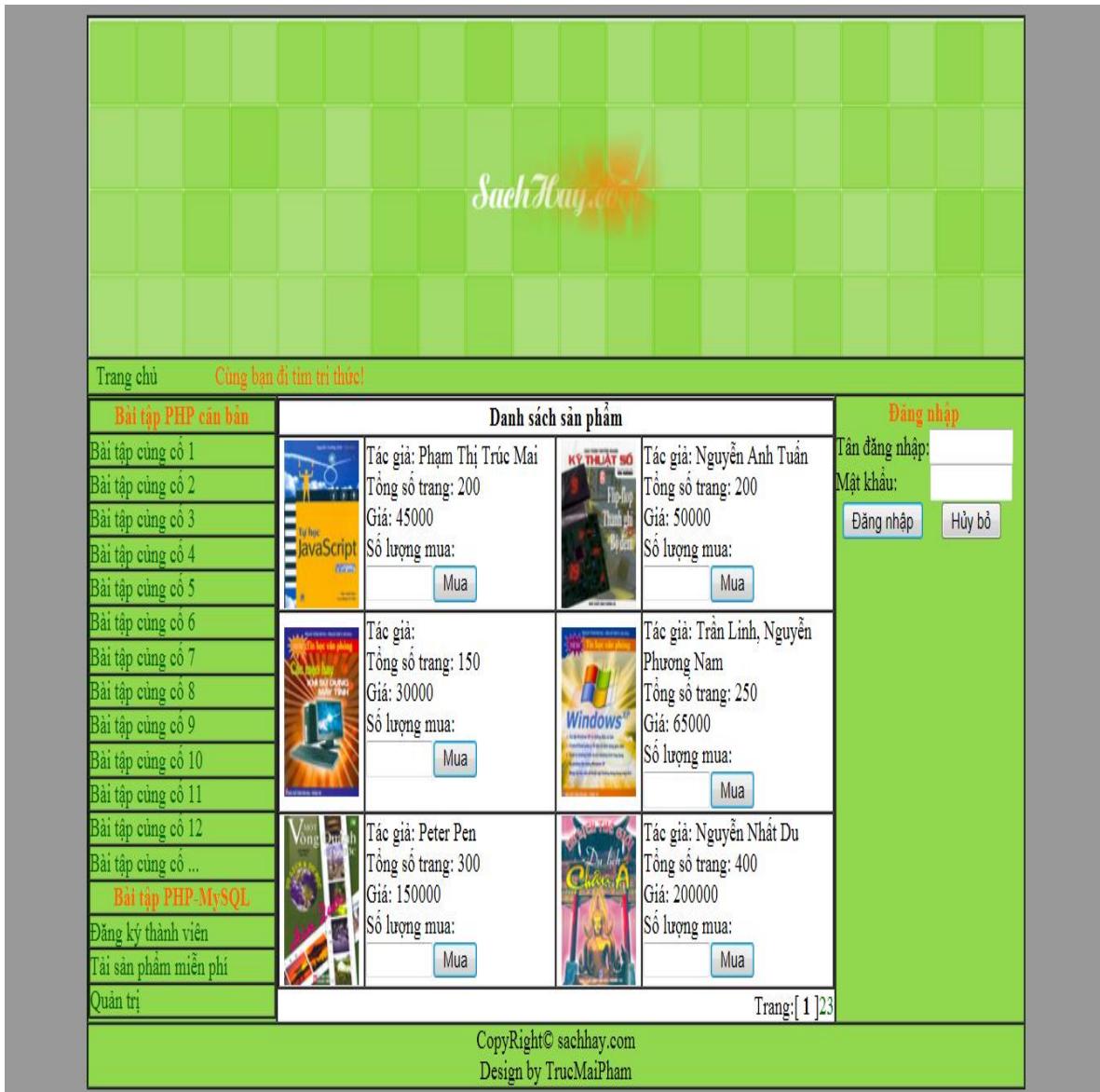
### HOÀN THÀNH MỘT TRANG WEB

❖ **Mục tiêu học tập:** Sau khi học xong bài này, người học có thể:

- Tự tạo một trang web cá nhân

#### 7.1. Yêu cầu

Xây dựng website bán sách trực tuyến có giao diện như hình bên dưới:



Hình 7-1: Trang Index.php

Đăng nhập quyền thành viên:

SachHay.com

Trang chủ	Cùng bạn đi tìm tri thức!	Xin chào	trucmai	Thoát			
Bạn đăng nhập với quyền thành viên của website							
<b>Danh sách sản phẩm</b>							
<a href="#">Bài tập PHP căn bản</a> <a href="#">Bài tập cùng cô 1</a> <a href="#">Bài tập cùng cô 2</a> <a href="#">Bài tập cùng cô 3</a> <a href="#">Bài tập cùng cô 4</a> <a href="#">Bài tập cùng cô 5</a> <a href="#">Bài tập cùng cô 6</a> <a href="#">Bài tập cùng cô 7</a> <a href="#">Bài tập cùng cô 8</a> <a href="#">Bài tập cùng cô 9</a> <a href="#">Bài tập cùng cô 10</a> <a href="#">Bài tập cùng cô 11</a> <a href="#">Bài tập cùng cô 12</a> <a href="#">Bài tập cùng cô ...</a> <a href="#">Bài tập PHP-MySQL</a> <a href="#">Chat với bạn bè</a> <a href="#">Tài sản phẩm miễn phí</a> <a href="#">Quản trị</a>	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 33%; padding: 5px; text-align: center;">             Tác giả: Phạm Thị Trúc Mai            Tổng số trang: 200            Giá: 45000            Số lượng mua:  <input type="button" value="Mua"/> </td> <td style="width: 33%; padding: 5px; text-align: center;">             Tác giả: Nguyễn Anh Tuấn            Tổng số trang: 200            Giá: 50000            Số lượng mua:  <input type="button" value="Mua"/> </td> <td style="width: 33%; padding: 5px; text-align: center;">             Tác giả: Trần Linh, Nguyễn Phương Nam            Tổng số trang: 250            Giá: 65000            Số lượng mua:  <input type="button" value="Mua"/> </td> </tr> <tr> <td style="width: 33%; padding: 5px; text-align: center;">             Tác giả: Peter Pen            Tổng số trang: 300            Giá: 150000            Số lượng mua:  <input type="button" value="Mua"/> </td> <td style="width: 33%; padding: 5px; text-align: center;">             Tác giả: Nguyễn Nhật Du            Tổng số trang: 400            Giá: 200000            Số lượng mua:  <input type="button" value="Mua"/> </td> <td style="width: 33%; padding: 5px;"></td> </tr> </table>	 Tác giả: Phạm Thị Trúc Mai Tổng số trang: 200 Giá: 45000 Số lượng mua: <input type="button" value="Mua"/>	 Tác giả: Nguyễn Anh Tuấn Tổng số trang: 200 Giá: 50000 Số lượng mua: <input type="button" value="Mua"/>	 Tác giả: Trần Linh, Nguyễn Phương Nam Tổng số trang: 250 Giá: 65000 Số lượng mua: <input type="button" value="Mua"/>	 Tác giả: Peter Pen Tổng số trang: 300 Giá: 150000 Số lượng mua: <input type="button" value="Mua"/>	 Tác giả: Nguyễn Nhật Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="button" value="Mua"/>	
 Tác giả: Phạm Thị Trúc Mai Tổng số trang: 200 Giá: 45000 Số lượng mua: <input type="button" value="Mua"/>	 Tác giả: Nguyễn Anh Tuấn Tổng số trang: 200 Giá: 50000 Số lượng mua: <input type="button" value="Mua"/>	 Tác giả: Trần Linh, Nguyễn Phương Nam Tổng số trang: 250 Giá: 65000 Số lượng mua: <input type="button" value="Mua"/>					
 Tác giả: Peter Pen Tổng số trang: 300 Giá: 150000 Số lượng mua: <input type="button" value="Mua"/>	 Tác giả: Nguyễn Nhật Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="button" value="Mua"/>						
Trang: [ 1 ] 23							
CopyRight© sachhay.com Design by TrucMaiPham							

Hình 7-2: Trang Index.php khi đăng nhập quyền thành viên

Khi đăng nhập với quyền quản trị:

SachHay.com				
Trang chủ	Cùng bạn đi tìm tri thức!			Xin chào
<a href="#">Bài tập PHP căn bản</a>	<a href="#">Danh sách sản phẩm</a>			<a href="#">Bạn đăng nhập với quyền admin</a>
<a href="#">Bài tập cùng cô 1</a>		Tác giả: Phạm Thị Trúc Mai Tổng số trang: 200 Giá: 45000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Anh Tuấn Tổng số trang: 200 Giá: 50000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Bài tập cùng cô 2</a>		Tác giả: Tổng số trang: 150 Giá: 30000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Trần Linh, Nguyễn Phương Nam Tổng số trang: 250 Giá: 65000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Bài tập cùng cô 3</a>		Tác giả: Tổng số trang: 150 Giá: 30000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Nhất Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Bài tập cùng cô 4</a>		Tác giả: Tổng số trang: 150 Giá: 30000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Nhất Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Bài tập cùng cô 5</a>		Tác giả: Tổng số trang: 150 Giá: 30000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Nhất Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Bài tập cùng cô 6</a>		Tác giả: Tổng số trang: 150 Giá: 30000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Nhất Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Bài tập cùng cô 7</a>		Tác giả: Tổng số trang: 150 Giá: 30000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Nhất Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Bài tập cùng cô 8</a>		Tác giả: Tổng số trang: 150 Giá: 30000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Nhất Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Bài tập cùng cô 9</a>		Tác giả: Tổng số trang: 150 Giá: 30000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Nhất Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Bài tập cùng cô 10</a>		Tác giả: Tổng số trang: 150 Giá: 30000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Nhất Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Bài tập cùng cô 11</a>		Tác giả: Tổng số trang: 150 Giá: 30000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Nhất Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Bài tập cùng cô 12</a>		Tác giả: Peter Pen Tổng số trang: 300 Giá: 150000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Nhất Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Bài tập cùng cô ...</a>		Tác giả: Peter Pen Tổng số trang: 300 Giá: 150000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Nhất Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Bài tập PHP-MySQL</a>		Tác giả: Peter Pen Tổng số trang: 300 Giá: 150000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Nhất Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Chat với bạn bè</a>				
<a href="#">Tài sản phẩm miễn phí</a>		Tác giả: Peter Pen Tổng số trang: 300 Giá: 150000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>		Tác giả: Nguyễn Nhất Du Tổng số trang: 400 Giá: 200000 Số lượng mua: <input type="text"/> <a href="#">Mua</a>
<a href="#">Quản trị</a>				Trang: [ 1 ] 23
<a href="#">Thêm sản phẩm mới</a>				
<a href="#">Sửa thông tin sách</a>				

CopyRight© sachhay.com  
Design by TrucMaiPham

Hình 7-3: Trang Index.php khi đăng nhập quyền quản trị

## **7.2. Phân tích**

Phần Bài tập PHP căn bản:

Chưa liên kết đến các bài tập củng cố từ chương 1 đến chương 3 mà sinh viên đã thực hiện được để từ trang giao diện của bài tập tổng hợp này, sinh viên có thể xem lại các bài tập trước đây một cách dễ dàng.

Phần Bài tập PHP – MySQL:

Giả lập một hệ thống bán sách trực tuyến. Khi duyệt Website như là khách, sẽ có liên kết cho phép khách đăng ký để trở thành thành viên của Website. Từ trang chủ này cũng có một liên kết dành cho quản trị viên. Khách có thể tải một số sản phẩm miễn phí được Website hỗ trợ.

Nếu là thành viên, bạn phải thông qua bước đăng nhập. Khi được xác định là thành viên của Website, bạn có quyền mua sách, sửa thông tin cá nhân, Chat với các thành viên khác hoặc với Ban quản trị...

Nếu là quản trị viên, bạn có quyền thêm sửa xóa thông tin sách, người dùng, cá nhân, Chat với các quản trị viên khác hoặc các thành viên trong hệ thống.

Về mặt lưu trữ, hệ thống cần các thông tin sau:

Thông tin thành viên: Mỗi thành viên có một tài khoản đăng nhập là duy nhất trên hệ thống, mật khẩu đăng nhập, họ tên thành viên, ngày sinh, giới tính, quốc tịch, hình ảnh đại diện.

Thông tin quản trị viên: tài khoản đăng nhập, mật khẩu, ....

Thông tin Chat: tài khoản của người gửi và người nhận, nội dung cuộc nói chuyện.

Thông tin sách: Mỗi đầu sách có một ID nhận dạng riêng, tên sách, mô tả hoặc tóm tắt nội dung, hình ảnh của cuốn sách để đăng lên web, đơn giá bán, tên tác giả, nhà xuất bản, số lượng hiện có trong kho, số trang, ....

Mỗi cuốn sách sẽ thuộc một loại hoặc lĩnh vực nào đó nhằm giúp khách hàng dễ dàng tìm kiếm.

## **7.3. Thiết kế**

Tên cơ sở dữ liệu: online\_shop

Bảng **admin**:

UserAdmin: varchar(20)

PassAdmin: text

Bảng **nguoidung**:

UserName: varchar(20)

PassWord: text

HoTen: varchar(50)

NgaySinh: date

GioiTinh: bit(2)

QuocTich: varchar(30)

HinhDaiDien: text

Bảng **chat**:

Id: int (auto\_increment)

NguoiGui: varchar(20)

NoiDung: text

NguoiNhan: varchar(20)

Bảng **loai**:

MaLoai: varchar(5)

TenLoai: varchar(50)

Bảng **sach**:

IdSach: int (auto\_encrement)

TenSach: varchar(100)

TacGia: varchar(100)

MoTa: text

HinhSach: text

SoTrang: int

Gia: int

SoLuong: int

MaLoai: varchar(5)

#### 7.4. Module dùng chung

Module kết nối CSDL

Module giao diện

Một số các hàm dùng chung khác (cải tiến sau)

#### 7.5. Các file dùng chung

File kết nối:

Tên file: ketnoi.php

Chức năng:

Kết nối đến máy chủ đang chứa cơ sở dữ liệu online\_shop

Lựa chọn cơ sở dữ liệu online\_shop

Nội dung file:

```
<?php  
$kn=mysql_connect("localhost","root","");
if(!$kn) die ("Không thể  
kết nối đến server". mysql_error());  
$csdl=mysql_select_db("online_shop", $kn);
if(!$csdl) die ("Không thể  
chọn được database online_shop". mysql_error());
mysql_query("SET NAMES 'utf8'", $kn);  
?>
```

#### Giao diện dùng chung:

Tên file: template.php, right\_and\_footer.php, index.php

Chức năng: Tạo phần giao diện chung cho các trang web.

Sinh viên vận dụng các kiến thức đã học để thiết kế trang index.php.

Phân tích: Chúng ta thấy rằng mỗi trang Web trên hệ thống Website chỉ khác nhau ở phần nội dung chính. Riêng cấu trúc là hoàn toàn giống nhau ở các trang. Khi đăng nhập ở các quyền khác nhau thì cấu trúc trang không đổi, chỉ thay đổi một vài nội dung bên trong

menu sao cho phù hợp với thực tế. Do đó, chúng tôi phân giao diện làm 3 phần: Phần Banner và menu trái (Phần 1); Phần chứa nội dung các trang (Phần 2); Phần đăng nhập và Footer của Website (Phần 3).

Tiến hành cắt giao diện:

Cắt phần 1 lưu vào trang mới và đặt tên file là template.php

Cắt phần 3 lưu vào trang mới và đặt tên file là right\_and\_footer.php

Phần 2 là phần còn lại nằm trên trang index.php

Tiến hành gọi lại giao diện cho trang index.php

Nội dung file index.php:

```
<?php
    include("template.php");
?>
    Nội dung chính của trang index
<?php
    include("right_and_footer.php");
?>
```

Tương tự, khi cần thiết kế một trang mới và cần có cấu trúc giao diện như trang index.php, bạn chỉ cần:

```
<?php
    include("template.php");
?>
    Nội dung chính của trang
<?php
    include("right_and_footer.php");
?>
```

## 7.6. Các chức năng

Hiển thị tất cả sách có trong hệ thống:

Tên file: index.php

Chức năng: Hiển thị tất cả các sách có trong cơ sở dữ liệu vào trang index.php.

Trang đăng ký thành viên:

Tên file: dangky.php

Chức năng: Cho phép đăng ký thành viên

Tên bảng truy xuất: nguoidung

Lệnh SQL thực hiện: insert

Giao diện:

The screenshot shows a web page titled "Đăng ký thành viên" (Member Registration) from the website SachHay.com. The page has a sidebar on the left containing a navigation menu with links like "Bài tập PHP căn bản", "Bài tập cùng có 1", etc. The main content area features a registration form with fields for "Tên đăng nhập", "Mật khẩu", "Nhập lại mật khẩu", "Họ và tên", "Giới tính" (with radio buttons for Nam and Nữ), "Ngày sinh" (birthdate), and "Quốc gia" (country, set to Việt Nam). Below the form is a section titled "Thông tin thêm" (Additional information) with a "Hình đại diện" (Profile picture) field and a "Browse..." button. A note in this section states: "Các điều khoản: Thành viên của website:  
- Viết tiếng Việt, có dấu rõ ràng.  
- Không bàn luận về chính trị.  
- Không lợi dụng tài khoản để tuyên truyền các loại hình văn hóa phẩm đồi trụy.  
Nếu vi phạm, tùy theo mức độ mà thành viên sẽ bị". There is also a checkbox labeled "Tôi đã đọc và chấp nhận các điều khoản trên!" (I have read and accepted the terms above!). At the bottom of the form are two buttons: "Đăng ký" (Register) and "Hủy bỏ" (Cancel). The footer of the page includes the copyright notice "CopyRight© sachhay.com" and "Design by TrucMaiPham".

**Hình 7-4: Trang đăng ký thành viên**

### Trang thông tin cá nhân:

Tên file: tt\_canhan.php

Chức năng: Hiển thị thông tin cá nhân của thành viên và cho phép chỉnh sửa.

Tên bảng truy xuất: nguoidung

Lệnh SQL thực hiện: select

Giao diện:

The screenshot shows a web page with a light green header featuring a logo. Below the header is a navigation bar with links: Trang chủ, Cùng bạn đi tìm tri thức!, Xin chào, truemai, and Thoát. The main content area has a title 'Thông tin của bạn'. It displays the following information:

- Họ và tên: Phạm Thị Trúc Mai
- Giới tính: Nữ
- Thuộc quốc tịch: Việt Nam
- Ảnh đại diện: (A small thumbnail of a pink rose flower.)

The left sidebar contains a vertical list of links:

- Bài tập PHP căn bản
- Bài tập cùng cỗ 1
- Bài tập cùng cỗ 2
- Bài tập cùng cỗ 3
- Bài tập cùng cỗ 4
- Bài tập cùng cỗ 5
- Bài tập cùng cỗ 6
- Bài tập cùng cỗ 7
- Bài tập cùng cỗ 8
- Bài tập cùng cỗ 9
- Bài tập cùng cỗ 10
- Bài tập cùng cỗ 11
- Bài tập cùng cỗ 12
- Bài tập cùng cỗ ...
- Bài tập PHP-MySQL** (This link is highlighted in red)
- Chat với bạn bè
- Tài sản phẩm miễn phí
- Quản trị

At the bottom of the page, there is a footer with the text: CopyRight© sachhay.com and Design by TrucMaiPham.

**Hình 7-5: Trang thông tin cá nhân**

### Trang tải sản phẩm:

Tên file: download.php

Chức năng: Hiển thị tất cả các file có trong thư mục Downloads/ và cho phép tải file.

Nội dung file:

```
<?php
$thumuc="downloads/";
$dd=opendir($thumuc);
echo("Đường dẫn download là: ".$thumuc);
echo("<br>Danh sách các file có thể download: <hr>");
while($taptin=readdir($dd))
{
    if($taptin != "." && $taptin != "..")
    {
        echo("<a href='".$thumuc.$taptin."'"
target='_blank'>".$taptin."</a>");
    }
}
closedir($dd);
?>
<?php
include("right_and_footer.php");
?>
```

Giao diện:

<i>Trí thức cho mọi nhà!!!!</i>			
Trang chủ	Cùng bạn đi tìm trí thức!	Xin chào truemai	Thoát
<a href="#">Bài tập PHP căn bản</a> <a href="#">Bài tập cùng cô 1</a> <a href="#">Bài tập cùng cô 2</a> <a href="#">Bài tập cùng cô 3</a> <a href="#">Bài tập cùng cô 4</a> <a href="#">Bài tập cùng cô 5</a> <a href="#">Bài tập cùng cô 6</a> <a href="#">Bài tập cùng cô 7</a> <a href="#">Bài tập cùng cô 8</a> <a href="#">Bài tập cùng cô 9</a> <a href="#">Bài tập cùng cô 10</a> <a href="#">Bài tập cùng cô 11</a> <a href="#">Bài tập cùng cô 12</a> <a href="#">Bài tập cùng cô ...</a> <a href="#">Bài tập DHD M&amp;CQT</a>	Đường dẫn download là: downloads/ Danh sách các file có thể download:  CHUONG 1_TONG QUAN VE LAP TRINH WEB DONG_new.pdf CHUONG 2_ON TAP HTML_new.pdf CHUONG 3_PHP.pdf CHUONG 4_PHP VA MYSQL.pdf CHUONG 5_QUA TRINH XAY DUNG WEBSITE DONG.pdf GIOI THIEU MON HOC_new.pdf So sanh Web tinh - Web dong.pptx	Bạn đăng nhập với quyền thành viên của website	

**Hình 7-6: Trang tải sản phẩm**

Trang quản trị:

Tên file: quantri.php

Chức năng:

Kiểm tra quyền quản trị.

Hiển thị thông tin tất cả người dùng trong hệ thống.

Tạo liên kết cho phép sửa / xóa thành viên

Tên bảng truy xuất: nguoidung

Lệnh SQL thực hiện: select

Giao diện:

Trang chủ	Cùng bạn đi tìm tri thức	Xin chào	admin	Thoát																			
<b>THÔNG TIN NGƯỜI DÙNG</b>																							
<a href="#">Bài tập PHP căn bản</a> <a href="#">Bài tập cùng cô 1</a> <a href="#">Bài tập cùng cô 2</a> <a href="#">Bài tập cùng cô 3</a> <a href="#">Bài tập cùng cô 4</a> <a href="#">Bài tập cùng cô 5</a> <a href="#">Bài tập cùng cô 6</a> <a href="#">Bài tập cùng cô 7</a> <a href="#">Bài tập cùng cô 8</a> <a href="#">Bài tập cùng cô 9</a> <a href="#">Bài tập cùng cô 10</a> <a href="#">Bài tập cùng cô 11</a> <a href="#">Bài tập cùng cô 12</a> <a href="#">Bài tập cùng cô ...</a> <a href="#">Bài tập PHP-MYSQL</a> <a href="#">Chat với bạn bè</a> <a href="#">Tài sản phẩm miễn phí</a> <a href="#">Quản trị</a> <a href="#">Thêm sản phẩm mới</a> <a href="#">Sửa thông tin sách</a>	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Tên đăng nhập</th> <th>Họ và tên</th> <th>Giới tính</th> <th>Quốc gia</th> <th>Hình đại diện</th> <th>Sửa</th> <th>Xóa</th> </tr> </thead> <tbody> <tr> <td>thanhhai</td> <td>Lưu Thanh Hải</td> <td>0</td> <td>Việt Nam</td> <td></td> <td><a href="#">Sửa</a></td> <td><a href="#">Xóa</a></td> </tr> <tr> <td>trucmai</td> <td>Phạm Thị Trúc Mai</td> <td>1</td> <td>Việt Nam</td> <td></td> <td><a href="#">Sửa</a></td> <td><a href="#">Xóa</a></td> </tr> </tbody> </table>	Tên đăng nhập	Họ và tên	Giới tính	Quốc gia	Hình đại diện	Sửa	Xóa	thanhhai	Lưu Thanh Hải	0	Việt Nam		<a href="#">Sửa</a>	<a href="#">Xóa</a>	trucmai	Phạm Thị Trúc Mai	1	Việt Nam		<a href="#">Sửa</a>	<a href="#">Xóa</a>	Bạn đăng nhập với quyền admin
Tên đăng nhập	Họ và tên	Giới tính	Quốc gia	Hình đại diện	Sửa	Xóa																	
thanhhai	Lưu Thanh Hải	0	Việt Nam		<a href="#">Sửa</a>	<a href="#">Xóa</a>																	
trucmai	Phạm Thị Trúc Mai	1	Việt Nam		<a href="#">Sửa</a>	<a href="#">Xóa</a>																	
CopyRight© sachhay.com																							

**Hình 7-7: Trang quản trị người dùng**

Trang sửa thông tin thành viên:

Tên file: sua\_thanhvien.php

Chức năng:

Kiểm tra quyền quản trị.

Hiển thị lại thông tin của một thành viên được chọn từ trang quantri.php

Ngoài tên đăng nhập, cho phép sửa tất cả các thông tin còn lại.

Thực hiện sửa thông tin.

Tên bảng truy xuất: nguoidung

Lệnh SQL thực hiện: update

Giao diện:

Trang chủ	Cùng bạn đi tìm tri thức!	Xin chào admin	Thoát
<a href="#">Bài tập PHP căn bản</a> <a href="#">Bài tập cùng cô 1</a> <a href="#">Bài tập cùng cô 2</a> <a href="#">Bài tập cùng cô 3</a> <a href="#">Bài tập cùng cô 4</a> <a href="#">Bài tập cùng cô 5</a> <a href="#">Bài tập cùng cô 6</a> <a href="#">Bài tập cùng cô 7</a> <a href="#">Bài tập cùng cô 8</a> <a href="#">Bài tập cùng cô 9</a> <a href="#">Bài tập cùng cô 10</a> <a href="#">Bài tập cùng cô 11</a> <a href="#">Bài tập cùng cô 12</a> <a href="#">Bài tập cùng cô ...</a>  <a href="#">Bài tập PHP-MySQL</a> <a href="#">Chat với bạn bè</a> <a href="#">Tài sản phẩm miễn phí</a> <a href="#">Quản trị</a> <a href="#">Thêm sản phẩm mới</a> <a href="#">Sửa thông tin sách</a>	Username: <input type="text" value="trucmai"/> Mật khẩu: <input type="password"/> Họ tên: <input type="text" value="Phạm Thị Trúc Mai"/> Giới tính: <input checked="" type="radio"/> Nam <input type="radio"/> Nữ Quốc gia <input type="text" value="Việt Nam"/>  Hình đại diện: <input type="text" value="Anh moi:"/> <input type="button" value="Browse..."/> <input type="button" value="Sửa thông tin"/> <input type="button" value="Hủy bỏ"/>	Bạn đã đăng nhập với quyền admin	
CopyRight© sachhay.com			

**Hình 7-8: Trang sửa thông tin thành viên**

Trang xóa thành viên:

Tên file: xoa\_thanhvien.php

Chức năng: Khi một thành viên vi phạm quy định của Ban quản trị Website, quản trị viên có quyền xóa thành viên đó ra khỏi hệ thống.

Tên bảng truy xuất: nguoidung

Lệnh SQL thực hiện: delete

Trang thêm sản phẩm mới:

Tên file: them\_sach.php

Chức năng:

Kiểm tra quyền quản trị.

Tạo form giao diện cho phép thêm thông tin của một đầu sách mới.

Thực hiện thêm sách mới.

Tên bảng truy xuất: sach

Lệnh SQL thực hiện: insert

Giao diện:

The screenshot shows a web-based administration interface for managing books. On the left, a sidebar lists various PHP topics. The main area is titled "Thêm sản phẩm mới" (Add new product). It contains input fields for "Tên sách" (Book name), "Tác giả" (Author), "Thuộc loại" (Category, with "Du lịch" selected), "Mô tả" (Description), "Hình sách" (Image, with a "Browse..." button), "Số trang" (Number of pages), "Giá" (Price), and "Số lượng" (Quantity). At the bottom are two buttons: "Thêm" (Add) and "Bỏ qua" (Cancel). The top right corner indicates the user is logged in as "admin".

**Hình 7-9: Trang thêm sản phẩm**

Trang quản trị nội dung sách:

Tên file: quantri\_sach.php

Chức năng:

Kiểm tra quyền quản trị.

Hiển thị thông tin của các quyền sách trong hệ thống

Tạo liên kết cho phép sửa / xóa một đầu sách.

Tên bảng truy xuất: sach, loai

Lệnh SQL thực hiện: select

Giao diện:

Trang chủ		Cùng bạn đi tìm tri thức!								Xin chào admin		Thoát	
		Thông tin sách từ CSDL								Bạn đăng nhập với quyền admin			
STT	Tên sách	Tác giả	Mô tả	Hình sách	Số trang	Đơn giá	Thuộc loại	Sửa	Xóa				
1	Tự học JavaScript	Phạm Thị Trúc Mai	Đây là quyển sách sẽ giúp bạn tự học các lệnh cũng như hiểu được cơ chế làm việc của JavaScript một cách đơn giản nhất...		200	45000	Công nghệ Thông tin	Sửa	Xóa				
2	Kỹ thuật số	Nguyễn Anh Tuấn	Quyển sách này giúp các bạn hiểu rõ về thanh ghi và bộ đệm cũng như cơ chế hoạt động, cách sửa chữa cơ bản		200	50000	Công nghệ Thông tin	Sửa	Xóa				
	Các mẹo		Giới thiệu một số phím tắt, các bước gửi cho máy tính luôn				Công						

**Hình 7-10: Trang quản trị nội dung sách**

Trang sửa một đầu sách:

Tên file: sua\_sach.php

Chức năng:

Kiểm tra quyền quản trị.

Hiển thị thông tin của một đầu sách.

Ngoài ID sách, cho phép sửa tất cả các thông tin còn lại của một đầu sách.

Thực hiện sửa thông tin.

Tên bảng truy xuất: sach

Lệnh SQL thực hiện: update

Giao diện:

Trang chủ		Cùng b Xin chào admin	Thoát
<a href="#">Bài tập PHP căn bản</a> <a href="#">Bài tập cùng cô 1</a> <a href="#">Bài tập cùng cô 2</a> <a href="#">Bài tập cùng cô 3</a> <a href="#">Bài tập cùng cô 4</a> <a href="#">Bài tập cùng cô 5</a> <a href="#">Bài tập cùng cô 6</a> <a href="#">Bài tập cùng cô 7</a> <a href="#">Bài tập cùng cô 8</a> <a href="#">Bài tập cùng cô 9</a> <a href="#">Bài tập cùng cô 10</a> <a href="#">Bài tập cùng cô 11</a> <a href="#">Bài tập cùng cô 12</a> <a href="#">Bài tập cùng cô ...</a> <a href="#">Bài tập PHP-MySQL</a> <a href="#">Chat với bạn bè</a> <a href="#">Tài sản phẩm miễn phí</a> <a href="#">Quản trị</a> <a href="#">Thêm sản phẩm mới</a> <a href="#">Sửa thông tin sách</a>		Sửa thông tin sách	Bạn đăng nhập với quyền admin
		<p>Id sách: <input type="text" value="1"/></p> <p>Tên sách: <input type="text" value="Tư học JavaScript"/></p> <p>Tác giả: <input type="text" value="Phạm Thị Trúc Mai"/></p> <p>Mô tả: Đây là quyển sách sẽ giúp bạn tự học các lệnh cũng như hiểu được cơ chế làm việc của JavaScript một cách đơn giản nhất...</p> <p>Hình sách: </p> <p>Chọn hình khác: <input type="text"/> <input type="button" value="Browse..."/></p> <p>Số Trang: <input type="text" value="200"/></p> <p>Đơn giá: <input type="text" value="45000"/></p> <p>Chọn loại sách: <input type="text" value="Công nghệ Thông tin"/></p> <p>Số lượng: <input type="text" value="10"/></p> <p><input type="button" value="Sửa"/> <input type="button" value="Làm lại"/></p>	
<a href="http://127.0.0.1/giaotrinhtw/bansach_online/BT5.php">http://127.0.0.1/giaotrinhtw/bansach_online/BT5.php</a>		CopyRight© sachhay.com	

**Hình 7-11: Trang sửa một đầu sách**

Trang xóa một đầu sách:

Tên file: xoa\_sach.php

Chức năng:

Kiểm tra quyền quản trị.

Xóa một đầu sách khi cần.

Tên bảng truy xuất: sach

Lệnh SQL thực hiện: delete.

Trang Chat với ban bè:

Tên file: ban\_chat.php

Chức năng:

Kiểm tra quyền thành viên hoặc quản trị.

Hiển thị các thành viên có trong hệ thống.

Cho phép chọn bạn Chat.

Tên bảng truy xuất: chat

Lệnh SQL thực hiện: select

Giao diện:

Trang chủ	Cùng bạn đi tìm tri Xin chào admin	Thoát										
Bài tập PHP căn bản Bài tập cùng cô 1 Bài tập cùng cô 2 Bài tập cùng cô 3 Bài tập cùng cô 4 Bài tập cùng cô 5 Bài tập cùng cô 6 Bài tập cùng cô 7 Bài tập cùng cô 8 Bài tập cùng cô 9 Bài tập cùng cô 10 Bài tập cùng cô 11 Bài tập cùng cô 12 Bài tập cùng cô ... <b>Bài tập PHP-MySQL</b> Chat với bạn bè Tài sản phẩm miễn phí Quản trị Thêm sản phẩm mới Sửa thông tin sách	<b>Chọn bạn Chat</b> <table border="1"><tr><td>Tên bạn Chat</td><td>Chọn</td></tr><tr><td>thanhhai</td><td>Chọn</td></tr><tr><td>trucmai</td><td>Chọn</td></tr></table> <b>Chọn người quản trị để Chat</b> <table border="1"><tr><td>Tên người quản trị</td><td>Chọn</td></tr><tr><td>quantrivien</td><td>Chọn</td></tr></table>	Tên bạn Chat	Chọn	thanhhai	Chọn	trucmai	Chọn	Tên người quản trị	Chọn	quantrivien	Chọn	Bạn đã đăng nhập với quyền admin
Tên bạn Chat	Chọn											
thanhhai	Chọn											
trucmai	Chọn											
Tên người quản trị	Chọn											
quantrivien	Chọn											
CopyRight© sachhay.com												

**Hình 7-12: Trang chọn bạn Chat**

Trang Chat:

Tên file: chat.php

Chức năng:

Kiểm tra quyền thành viên hoặc quản trị.

Hiển thị nội dung Chat giữa người đang đăng nhập và bạn Chat.

Cho phép thêm nội dung mới

Tên bảng truy xuất: chat

Lệnh SQL thực hiện: select, insert

Giao diện:

The screenshot shows a web-based chat application interface. At the top, there is a navigation bar with links: 'Trang chủ' (Home), 'Cùng bạn đi tìm tri thức!' (Together with you to find knowledge!), 'Xin chào' (Hello), 'trucmai' (username), and 'Thoát' (Logout). Below the navigation bar is a sidebar on the left containing a list of links under two main categories: 'Bài tập PHP căn bản' (Basic PHP exercises) and 'Bài tập PHP-MySQL'. The 'Bài tập PHP căn bản' section includes links from 'Bài tập cùng cỗ 1' to 'Bài tập cùng cỗ ...'. The 'Bài tập PHP-MySQL' section includes links for 'Chat với bạn bè', 'Tài sản phẩm miễn phí', and 'Quản trị'. The main content area on the right displays a conversation between 'admin' and 'trucmai'. The messages are:  
admin Gửi tới trucmai: bạn cứ nói, nếu biết sẽ giải đáp liền  
trucmai Gửi tới admin: mình có một thắc mắc muốn hỏi  
admin Gửi tới trucmai: chào bạn  
trucmai Gửi tới admin: chào admin,  
Below the conversation is a text input field labeled 'Nhập nội dung:' and a blue 'Gửi' (Send) button. At the bottom of the page, there is a footer with the text 'CopyRight© sachhay.com'.

Hình 7-13: Trang Chat trực tuyến

**❖ Câu hỏi (bài tập) củng cố:**

1. Hoàn thành trang web theo hướng dẫn mẫu
2. Xây dựng diễn đàn đơn giản. Trong ứng dụng cần thể hiện được các tiêu chí sau:
  - Trong diễn đàn sẽ phân ra các loại: Lĩnh vực, chủ đề và lời bình luận.
  - Người quản trị có quyền thêm lĩnh vực, tạo chủ đề và ghi lời bình luận
  - Người dùng đã đăng nhập thành công thì có quyền chọn lĩnh vực cần xem, tạo chủ đề và ghi lời bình luận
  - Người dùng chưa đăng nhập thì chỉ có quyền xem lĩnh vực, xem chủ đề và xem lời bình luận.
  - Trong mỗi chủ đề và lời bình luận cần hiển thị cho biết ai tạo, và tạo khi nào.
  - Mối quan hệ giữa các bảng là:
    - + Mỗi lĩnh vực sẽ có nhiều chủ đề được tạo ra; mỗi chủ đề chỉ thuộc một lĩnh vực.
    - + Mỗi chủ đề sẽ có nhiều lời bình luận; mỗi lời bình luận thuộc một chủ đề.

## PHỤ LỤC

### CÔNG CỤ HỖ TRỢ THIẾT KẾ WEB

#### A. Ý nghĩa và một số công cụ hỗ trợ thiết kế Web

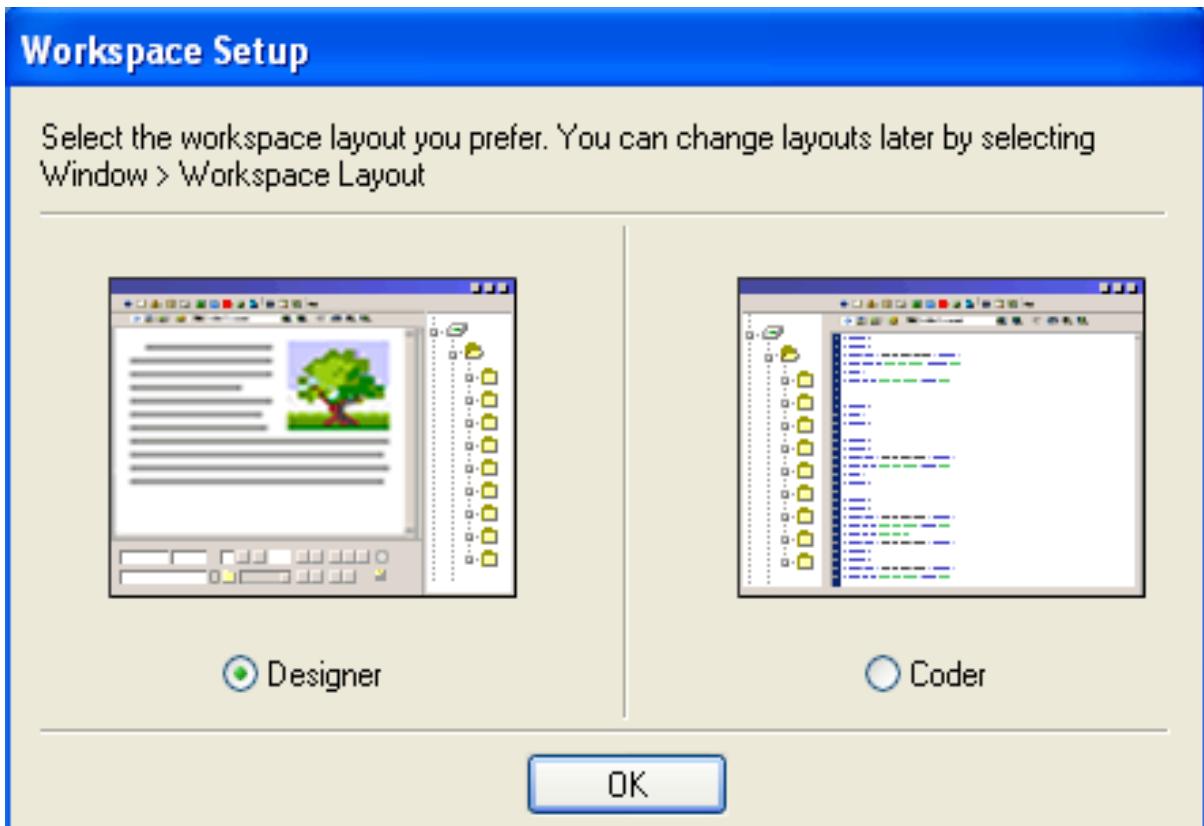
Như đã biết, chúng ta có thể soạn thảo một trang HTML bằng nhiều chương trình từ cơ bản như Notepad, Wordpad, Microsoft Office Word cho đến các chương trình soạn thảo Web chuyên nghiệp như Microsoft FrontPage, Macromedia Dreamweaver, Adobe Dreamweaver, ... Tuy nhiên, thay vì phải gõ từng từ khóa, nhớ chính xác từng thuộc tính của thẻ khi soạn thảo bằng các chương trình soạn thảo cơ bản, những nhà phát triển Web thường chọn cho mình một chương trình hỗ trợ soạn thảo Web chuyên nghiệp để không mất quá nhiều thời gian và công sức cho việc viết mã lệnh.

Trong quyển tài liệu tham khảo này, nhóm biên soạn chúng tôi xin giới thiệu đến các bạn công cụ hỗ trợ soạn thảo thiết kế Web của có tên Dreamweaver. Hiện tại phần mềm này được sản xuất bởi Macromedia và Adobe. Đây là một công cụ rất mạnh, hỗ trợ thiết kế Web phổ biến hiện nay. Chương trình này hỗ trợ soạn thảo rất nhiều kiểu file như: HTML, CSS, JavaScript, PHP, XML, ASP.NET C#, ASP.NET VB,... Bên cạnh đó nó còn hỗ trợ nhiều bảng kiểu (giao diện – Layout) có sẵn.



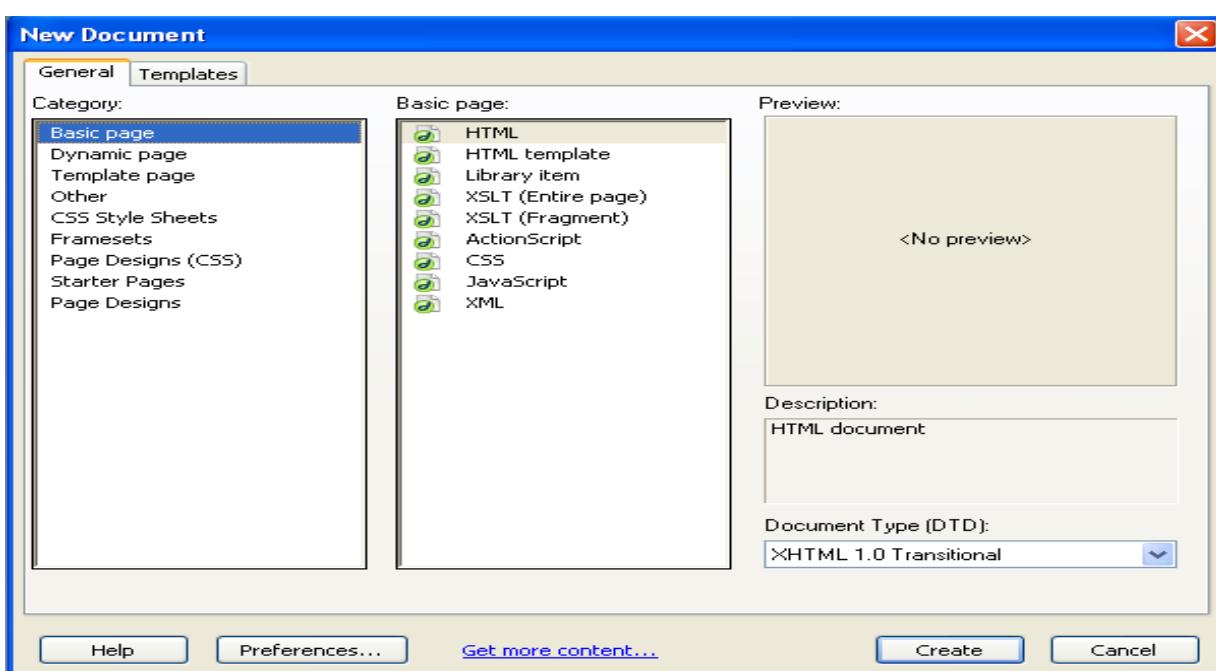
Biểu tượng của chương trình:

Trong quá trình khởi động, chương trình sẽ cho phép bạn chọn chế độ soạn thảo dạng thiết kế với chuột (Design) hay với mã lệnh (Code).



**Hình PL-1: Màn hình khởi động – chọn cửa sổ làm việc mà bạn mong muốn**

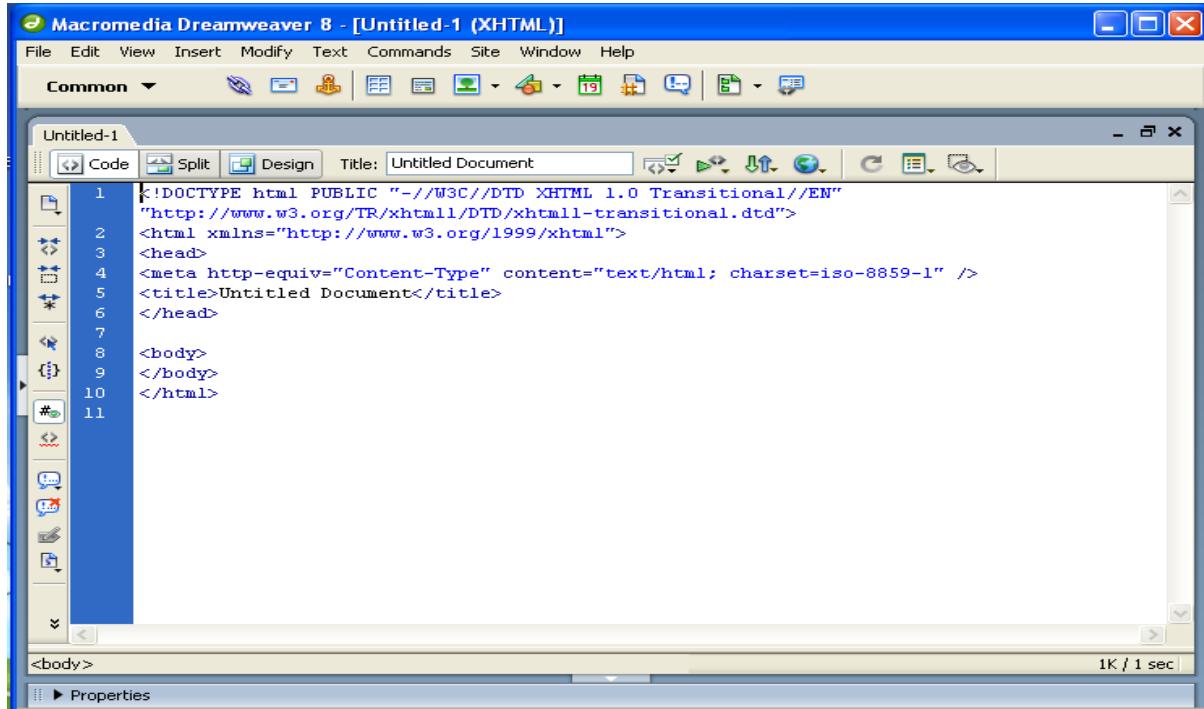
Bạn có thể chọn ngôn ngữ mà mình sẽ soạn thảo khi chọn File → New



**Hình PL-2: Chọn kiểu file**

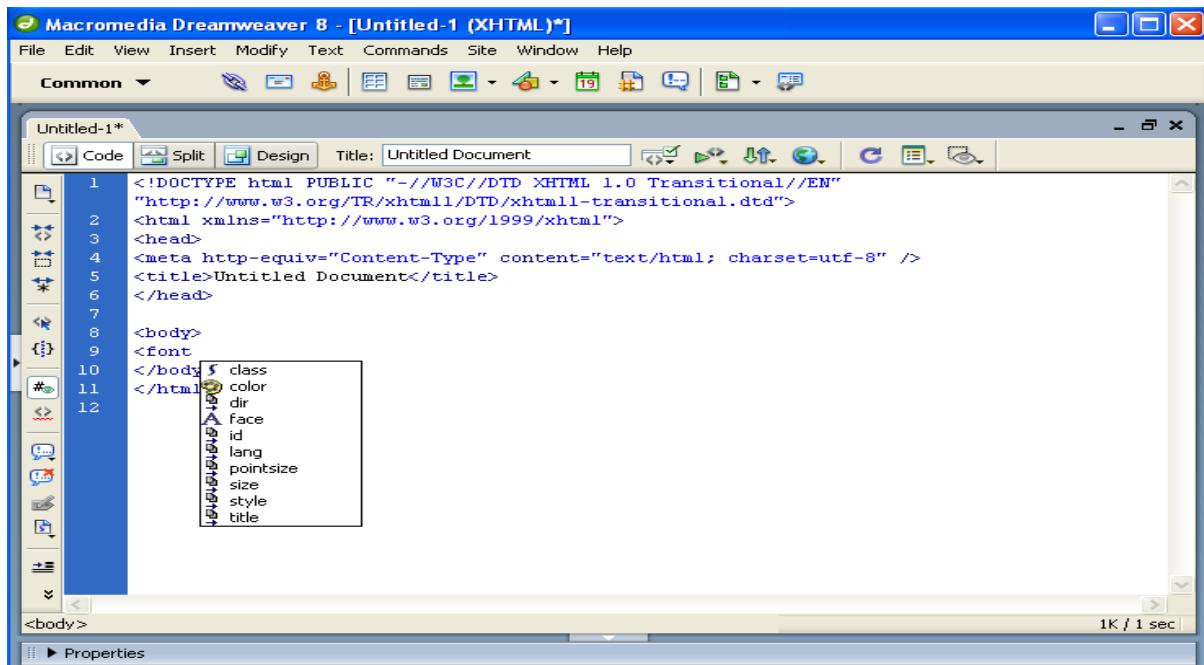
## B. Một số chức năng hỗ trợ đáng chú ý

Dreamweaver hỗ trợ ba dạng cửa sổ làm việc: Code, Split và Design.



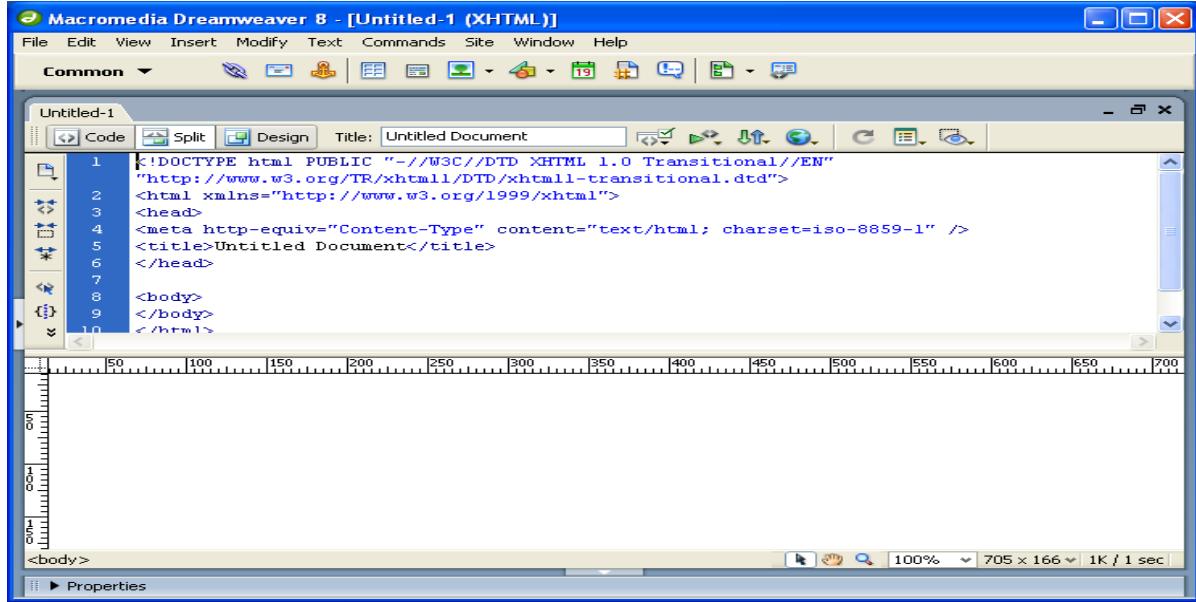
Hình PL-3: Cửa sổ Code

Với cửa sổ Code, Dreamweaver sẽ hiển thị gợi ý khi bạn gõ dấu mở thẻ.



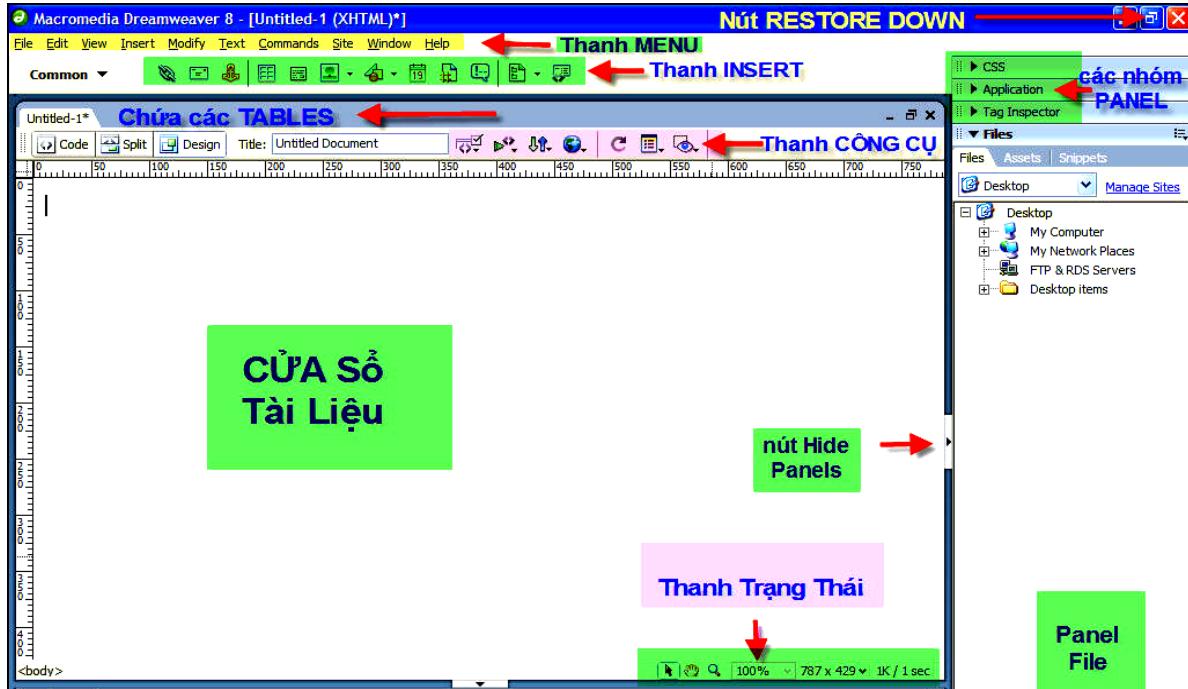
Hình PL-4: Dreamweaver hỗ trợ code

Chế độ Split sẽ chia cửa sổ làm việc thành hai phần, bạn có thể vừa soạn thảo code vừa xem kết quả thiết kế.



Hình PL-5: Chế độ Split

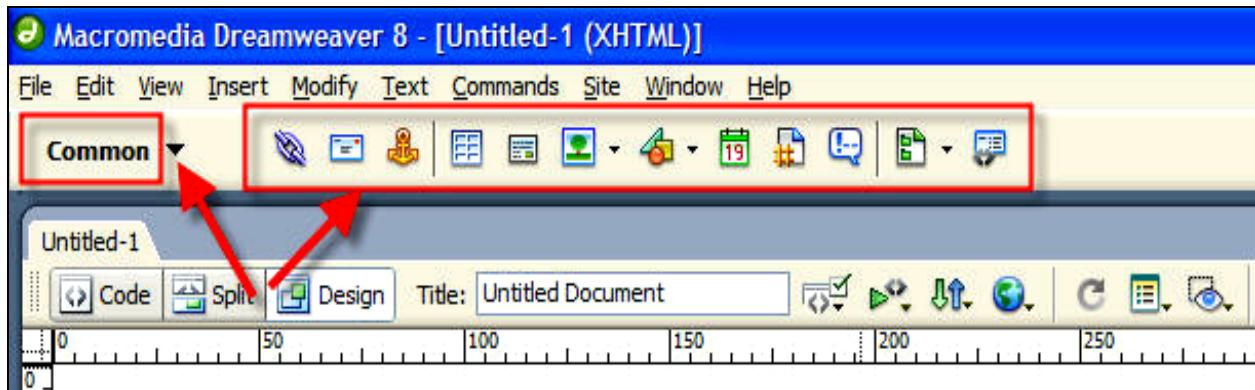
Chế độ Design, ngoài việc gõ nội dung trang Web, bạn không cần nhớ mã HTML. Thay vào đó bạn phải biết cách gọi các thẻ bằng cách chọn lệnh tương ứng từ menu.



Hình PL-6: Chế độ Design

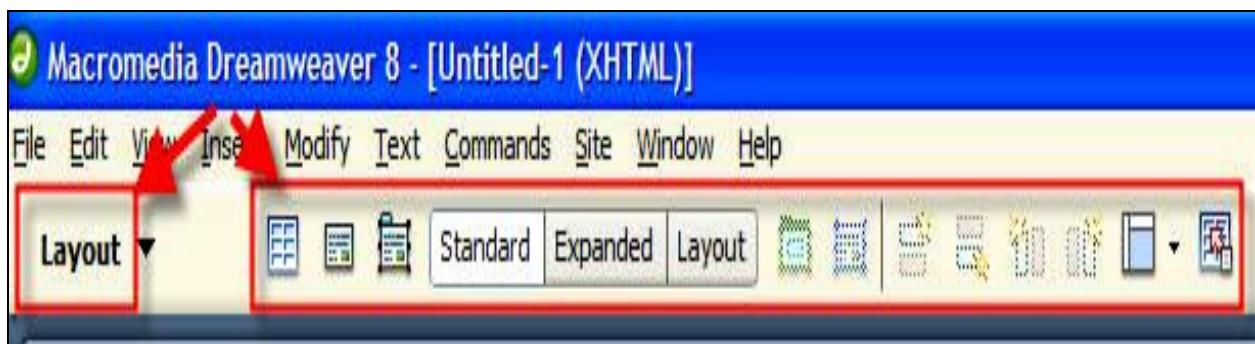
Sau đây, chúng ta sẽ tìm hiểu một số thanh công cụ của Dreamweaver.

Thanh Common, chứa các đối tượng thường được sử dụng phổ biến nhất như các liên kết với ảnh.



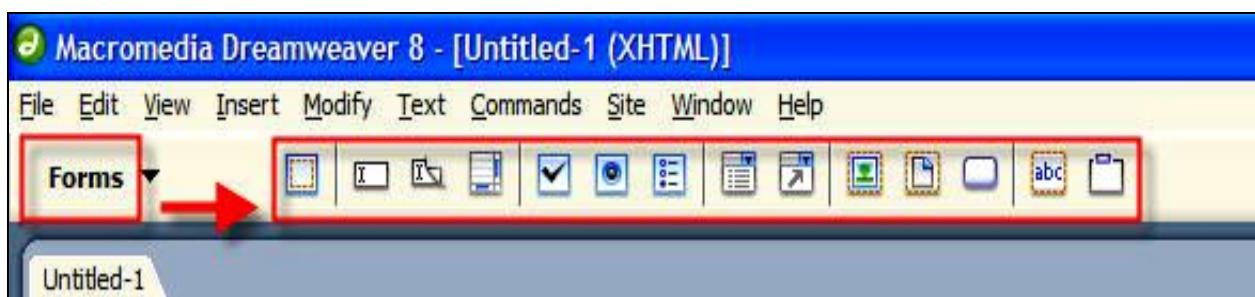
**Hình PL-7: Thanh Common**

Thanh Layout, giúp bạn mô tả cách bạn muốn trình bày trang: table, div, frame.



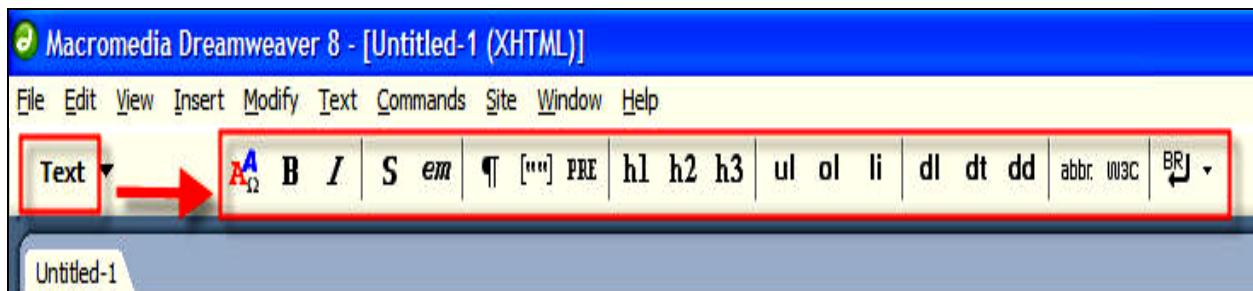
**Hình PL-8: Thanh Layout**

Thanh Form, gồm các thành phần Form như trường Text, nút lệnh và hộp chọn, ...



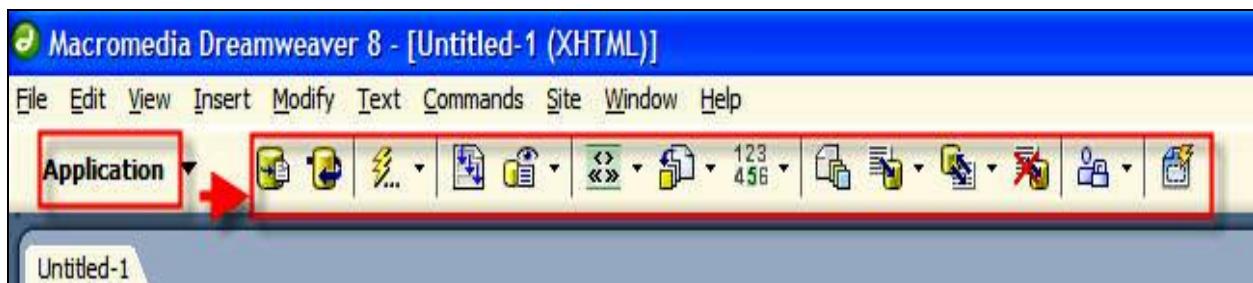
**Hình PL-9: Thanh form**

Thanh Text, hỗ trợ định dạng văn bản bằng nút lệnh.



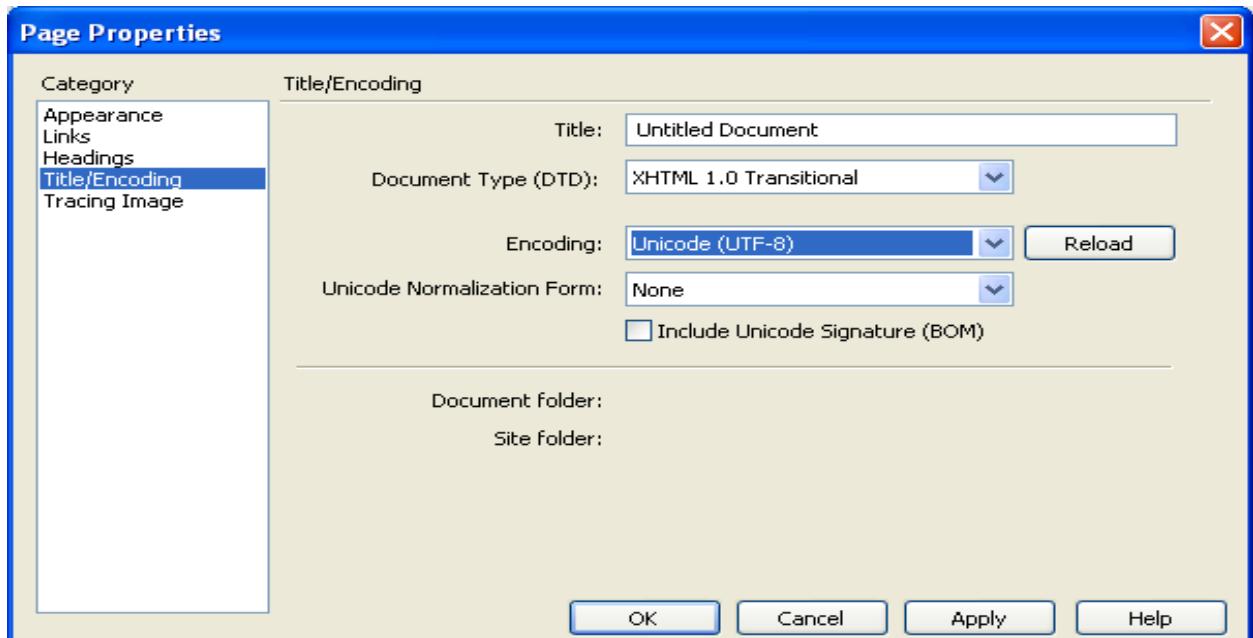
Hình PL-10: Thanh Text

Thanh Application, giúp cho bạn làm việc với các cơ sở dữ liệu bên ngoài.



Hình PL-11: Thanh Application

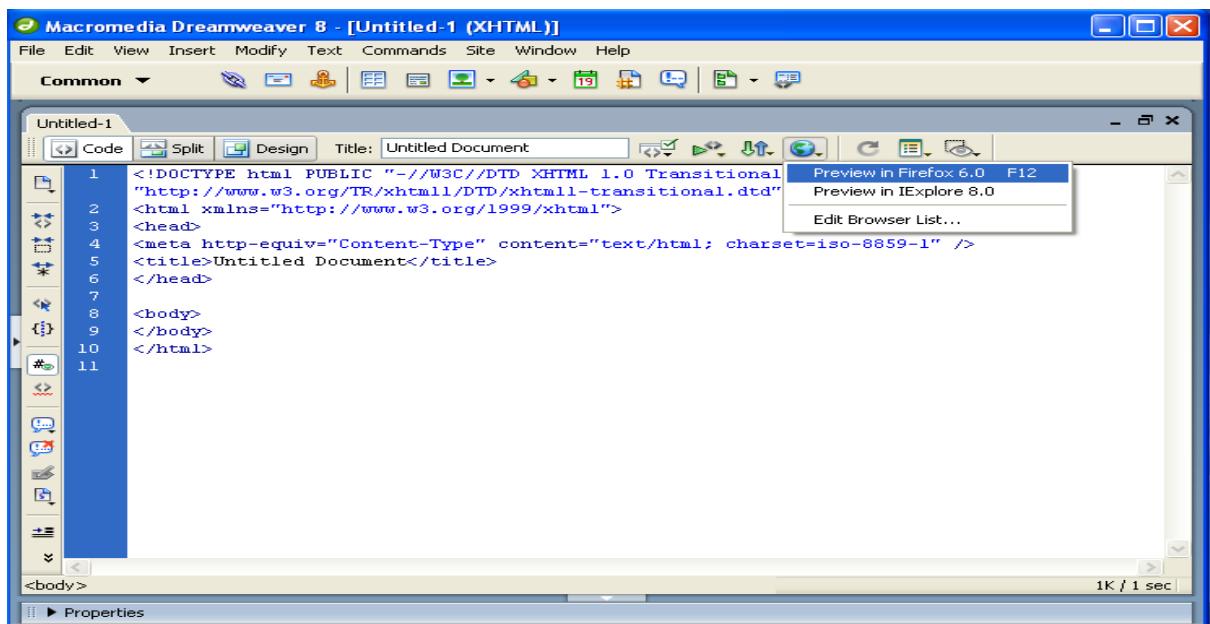
Để soạn thảo và hiển thị đúng tiếng Việt, bạn vào Modify → Page Properties → Title-Encoding và chọn trong Unicode (UTF-8) trong mục Encoding. Sau đó Apply và OK để thiết lập của bạn có hiệu lực ngay.



Hình PL-12: Hiệu chỉnh Encoding

### C. Xuất bản kết quả

Khi muốn kiểm tra kết quả thiết kế của mình trên trình duyệt, bạn chỉ cần lưu tập tin lại lần cuối trước khi Test. Sau đó từ cửa sổ soạn thảo, bạn chọn vào biểu tượng Preview bằng trình duyệt (có hình quả địa cầu) và chọn trình duyệt kiểm thử.



Hình PL-13: Xuất bản kết quả

## TÀI LIỆU THAM KHẢO

### ❖ **TÀI LIỆU THAM KHẢO ĐỂ BIÊN SOẠN NỘI DUNG MÔN HỌC:**

[1] - Xây dựng ứng dụng Web bằng PHP & MySQL - Phạm Hữu Khang, NXB Phương Đông, 2007.

[2] - Bài giảng môn Lập trình Web, phần HTML, JavaScript – Đỗ Thanh Nghị - Khoa Công nghệ Thông tin và Truyền thông, Trường Đại học Cần Thơ.

[3] - Bài giảng môn Thiết kế Web - Phạm Thị Trúc Mai – Bộ môn Công nghệ Thông tin, Khoa Kỹ thuật & Công nghệ, Trường Đại học Trà Vinh.

[4] - Đào Việt Cường, Bài giảng Lập trình web động PHP, Khoa CNTT – ĐHSP Hà Nội.

[5] - Bài giảng Lập trình web PHP, Trung tâm tin học Trường Đại Học KHTN Thành Phố Hồ Chí Minh.

[6] - Tài liệu CSS cơ bản của tác giả WallPearl, địa chỉ Email: [wallpearl@gmail.com](mailto:wallpearl@gmail.com).

[7] - <http://www.w3schools.com/books/default.asp>

[8] - <http://info.cern.ch>

[9] - <http://www.wikipedia.org>

[10] - <http://www.java2s.com/Code/Php/CatalogPhp.htm>

[11] - <http://www.w3schools.com/php/default.asp>

[12] - <http://www.itelvn.net/threads/series-php-bai-4-mang-trong-php.131/>

### ❖ **TÀI LIỆU THAM KHẢO ĐỂ NGHỊ CHO HỌC VIÊN:**

[1] - Xây dựng ứng dụng Web bằng PHP & MySQL - Phạm Hữu Khang, NXB Phương Đông, 2007.

[2] - <http://www.w3schools.com/books/default.asp>

[3] - <http://www.java2s.com/Code/Php/CatalogPhp.htm>

[4] - <http://www.w3schools.com/php/default.asp>