

Tổng Hợp Mọi Thứ Về PHP (TEST)

Mục Lục

- Chapter 1 : Giới Thiệu PHP
- Chapter 2 : Bắt Đầu Với PHP
- Chapter 3 : Cài Đặt PHP
- Chapter 4 : Cú Pháp PHP
- Chapter 5 : Biến PHP
- Chapter 6 : Chuỗi Biến PHP
- Chapter 7 : Toán Tử PHP
- Chapter 8 : Câu lệnh if ... else trong PHP .
- Chapter 9 : Câu lệnh Switch trong PHP
- Chapter 10 : Arrays (Mảng)
- Chapter 11 : Looping – While Loops
- Chapter 12 : Looping – For Loops
- Chapter 13 : Functions (Hàm Chức Năng)
- Chapter 14 : PHP Forms and User Input
- Chapter 15 : PHP \$_GET Variable
- Chapter 16 : PHP \$_POST Function
- Chapter 17 : Date() Function
- Chapter 18 : File Handling
- Chapter 19 : File Upload
- Chapter 20 : PHP Cookies
- Chapter 21 : PHP Sessions
- Chapter 22 : Sending Emails
- Chapter 23 : Secure E-mails
- Chapter 24 : Error Handling
- Chapter 25 : Exception Handling
- Chapter 26 : PHP Filter
- Chapter 27 : PHP MySQL
- Chapter 28 : PHP XML Expat Parser
- Chapter 29 : Giới thiệu AJAX
- Chapter Ngoài : Bài Tập
- Bài 11: Viết ứng dụng đăng nhập bằng PHP và MYSQL
- Bài 12: Viết ứng dụng thêm thành viên bằng PHP và MYSQL
- Bài 13: Viết ứng dụng quản lý thành viên bằng PHP và MYSQL
- Bài 14: viết ứng dụng sửa xóa thành viên bằng PHP và MYSQL
- Bài 15: Viết ứng dụng tạo mã xác nhận bằng PHP
- Bài 16: Viết ứng dụng đếm số người online bằng php

- Bài 17: Xây dựng hệ thống bình chọn bằng PHP và MYSQL
- Bài 18: Xây dựng ứng dụng tạo trang upload nhiều hình ảnh
- Bài 19: Viết ứng dụng tạo giỏ hàng - shopping cart - phần 1
- Bài 20: Viết ứng dụng tạo giỏ hàng - shopping cart - phần 2
- Chapter 30 : Các hàm
- Chapter 31 : Linh Tinh Các Thứ Bên Ngoài Tổng Hợp Được
- +Code style theo chuẩn Zend
- Những điều cần biết về PHP 5.4
- Các hàm rất hữu ích trong PHP bạn cần biết !
- Điều gì khiến bạn là 1 coder tồi ?
- Các lỗi cơ bản trong lập trình PHP
- Code gửi mail trong PHP
- Tự làm menu 2 cấp đơn giản với php và javascript
- Tự làm điểm sáng với javascript
- Giới thiệu - Cài đặt Smarty
- Smarty cơ bản [Phần 1]
- Smarty cơ bản [Phần 2]
- Kỹ thuật sử dụng lazy loading trong hướng đối tượng
- Regular Expressions
- Giới thiệu Regular Expression Nâng cao
- Đệ quy với menu không giới hạn số cấp (N cấp)
- Upload file trong php
- Class support Mutil Language cho cơ bản và nâng cao và cho cả các framework khác!!!!
- Hướng dẫn CSS Framework 960 cho mọi người
- Nhập dữ liệu từ file excel sử dụng công nghệ XML...
- Mô hình VMC auto load Controller và model
- Thủ Thuật Tăng Tốc PHP
- Code phân trang theo kiểu Google
- Phân trang cơ bản, có phân đoạn không OOP
- Hướng dẫn sử dụng jquery ajax lồng nhau trong ứng dụng
- Code phân trang bằng Ajax đơn giản
- Sử dụng JqGrid PHP
- Kĩ thuật seo website
- Phân trang đơn giản với JQuery
- Performance Tips trong PHP - Part 1
- Những lớp và phương thức trừu tượng trong PHP 5.
- Mô hình Multi Modules , Multi Templates và Multi Database
- Tìm hiểu thư viện Php_filter
- Viết Class Database có chống SQL Injection
- Cách lấy rss và tin tức Từ VnExpress
- Cách dùng hàm ereg, preg trong php
- Khi dùng ajax để khi nhấn enter sẽ tự động gọi hàm giống khi dùng form
- [TUT] Add dữ liệu từ form vào database hiển thị tiếng việt có dấu.
- Tản mạn về ECHO
- Code xem thư mục con .
- STRING trong PHP [PART1]

- STRING trong PHP [PART2]
- STRING trong PHP [PART3]
- STRING trong PHP [PART4]
- Mảng trong PHP [Part1]
- Mảng trong PHP [Part2]
- Hướng dẫn code trang news cơ bản
- HƯỚNG DẪN CODE TRANG BÁN SÁCH
- Hướng dẫn chi tiết code trang bán sách

About Author of this book

Author : Siverdragon12

- Vì lý do ngẫu hứng và trong lúc học lập trình php nên muốn viết lại quyển sách này dành cho người mới nhập môn hoặc đa phần các bạn không biết tiếng anh ... vì 1 tương lai ai cũng giỏi CNTT nhĩ ... không có nghĩa ai không biết tiếng anh thì không thể lập trình .
- Have A Nice Day , everybody !!!

Chú Ý : Sách Chưa được hoàn thiện . Đang được bổ sung và chỉnh sửa nhiều thứ để giúp sách ngày một hoàn thiện hơn vì vậy sách hiện đang là bản test , sẽ sớm public bản hoàn thiện và thêm nhiều chapter mới , bài tập mới , training mới cho các bạn và thêm phần nâng cao PHP để giúp các bạn hiểu hơn và lập trình tốt hơn với PHP . Share To Be Shared . Cảm ơn .

Chapter 1 : Giới Thiệu PHP

PHP là một ngôn ngữ kịch bản máy chủ, và là một công cụ mạnh để làm cho các trang Web động và tương tác.

PHP là một thay thế được sử dụng rộng rãi, miễn phí, và hiệu quả với các đối thủ cạnh tranh như ASP của Microsoft.

"Show PHP" công cụ của chúng tôi làm cho nó dễ dàng để tìm hiểu PHP, nó cho thấy cả mã nguồn PHP và đầu ra của mã HTML.

Ví Dụ 1 Đoạn Code PHP đơn giản kết hợp HTML

```
<!DOCTYPE html>
<html>
<body>

<?php
echo "My first PHP script!";
?>

</body>
</html>
```

Phần Đỏ Đậm là ngôn ngữ PHP , còn lại là HTML ☺

Chapter 2 : Bắt đầu với PHP

Những gì bạn nên biết !!!

Trước khi tiếp tục, bạn cần phải có một sự hiểu biết cơ bản sau đây:

- HTML
- Javascript

Nếu bạn muốn nghiên cứu các đối tượng này lần đầu tiên, tìm thấy các hướng dẫn trên trang này .

<http://w3schools.com/>

PHP là gì?

- PHP là viết tắt của PHP: Hypertext Preprocessor
- PHP là một ngôn ngữ sử dụng rộng rãi , là một mã nguồn mở
- PHP được thực hiện trên máy chủ (vd : Linux)
- PHP hoàn toàn miễn phí để download và sử dụng

PHP là ngôn ngữ đơn giản nhất cho người mới bắt đầu học lập trình

PHP cũng cung cấp nhiều tính năng tiên tiến cho các lập trình viên chuyên nghiệp.

File PHP là gì?

- PHP tập tin có thể chứa văn bản, HTML, JavaScript mã, và mã PHP
- Mã PHP được thực hiện trên máy chủ, và kết quả được trả về cho trình duyệt như

HTML

- PHP tập tin có phần mở rộng tập tin mặc định của ". Php"

PHP có thể làm gì?

- PHP có thể tạo ra nội dung trang năng động
- PHP có thể tạo ra, mở, đọc, viết, và các tập tin trên máy chủ gần
- PHP có thể thu thập dữ liệu mẫu
- PHP có thể gửi và nhận cookie
- PHP có thể thêm, xóa, sửa đổi dữ liệu trong cơ sở dữ liệu của bạn
- PHP có thể hạn chế người dùng truy cập một số trang trên trang web của bạn
- PHP có thể mã hóa dữ liệu

Với PHP, bạn không giới hạn đầu ra HTML. Bạn có thể xuất hình ảnh, tập tin PDF, và thậm chí cả phim Flash. Bạn cũng có thể sản xuất bất kỳ văn bản, chẳng hạn như XHTML và XML.

Tại sao PHP?

- PHP chạy trên các nền tảng khác nhau (Windows, Linux, Unix, Mac OS X, vv)
- PHP là tương thích với hầu như tất cả các máy chủ được sử dụng hiện nay (Apache, IIS, vv)
- PHP đã hỗ trợ cho một loạt các cơ sở dữ liệu
- PHP là miễn phí. Tải về nó từ nguồn PHP chính thức: www.php.net
- PHP là dễ dàng để tìm hiểu và chạy hiệu quả ở phía máy chủ

Chapter 3 : Cài đặt PHP

Những gì bạn cần có !

- + Tìm 1 Hosting (để chạy php , cái này áp dụng cho ai muốn thuê host bỏ tiền mình khuyên nên chạy localhost)
- + Bạn có thể dùng các chương trình hỗ trợ chạy localhost trên google , có rất nhiều và bạn có thể tùy chọn nếu thích ☺

Đối với việc Chọn Hosting

- Bạn nên chọn hosting nào hỗ trợ tốt về tư vấn khách hàng (vì đó là yếu tố cần thiết)
- Bạn nên chọn hosting nào có tốc độ load nhanh , hỗ trợ đường truyền tốt và không bị cấm các hàm (Function) để tạo điều kiện tốt nhất cho việc học PHP

Các chương trình chạy Localhost đều phải có các điều sau đây

- Có hệ thống Apache
- MySQL
- PHP

Đó là những yêu cầu cơ bản của 1 server (máy chủ)

Chapter 4 : Cú Pháp PHP

Một số cú pháp đơn giản

Tất cả các mã lệnh php đều bắt đầu với `<?php` và kết thúc với `?>` . Mã lệnh của PHP có thể đặt ở bất cứ đâu trong file .

Ở 1 số máy chủ khác bạn cũng có thể bắt đầu với `<?` Và kết thúc với `?>`

Đối với khả năng tương thích tối đa, chúng tôi khuyên bạn nên sử dụng các hình thức tiêu chuẩn (`<? Php`) chứ không phải là hình thức viết tắt.

Ví Dụ

```
<?php  
Mã lệnh php sẽ ở đây  
?>
```

Phần mở rộng tập tin mặc định cho các tập tin PHP là ".php"

Một file PHP thông thường có chứa các thẻ HTML, và một số mã kịch bản PHP.

Dưới đây, chúng tôi có một ví dụ của một kịch bản PHP đơn giản mà sẽ gửi văn bản "Hello World!" cho trình duyệt:

```
<?php  
echo "Hello World!";  
?>
```

Mỗi dòng mã trong PHP phải kết thúc bằng một dấu chấm phẩy. Dấu **chấm phẩy** là một dấu phân cách và được sử dụng để phân biệt một bộ các hướng dẫn khác.

Có hai câu lệnh cơ bản để văn bản đầu ra với PHP là : **echo** và **print**.

Trong ví dụ trên, chúng tôi đã sử dụng **echo** tuyên bố để sản xuất các văn bản "Hello World".

Chú Thích Trong PHP

Trong PHP , Chúng ta sử dụng `//` (Không có dấu ngoặc kép) để làm 1 dòng làm chú thích , hoặc `/*` và `*/` để làm chú thích dạng khối

Ví dụ :

```
<!DOCTYPE html>
<html>
<body>

<?php
//Đây là chú thích 1 dòng

/*
Đây là chú thích
Nhiều
Dòng
*/
?>

</body>
</html>
```

Chapter 5 : Biến PHP

Bạn còn nhớ môn Đại Số ở Trường Không ?

Dạng như , $x = 1$, $y = 2$, $z = 3$

Bạn còn nhớ với mỗi từ ta có thể gán 1 giá trị (vd : $x = 2$, $y = 123$, $z = 52$ v.v) và bạn sẽ dùng mấy thông tin đó để tính 1 giá trị nào đó của d chẳng hạn .

Tất cả những cái trên được gọi là **Biến PHP** , và biến được dùng để giữ 1 giá trị nhất định ($x=2$) hoặc là các bài toán như ($d = a+b+c$) trong đó a,b,c là các hằng số có giá trị bất kỳ (vd : $a = 1$,

$b = 2$, $c = 3$ thì $d = a + b + c = 6$)

Biến PHP

Với đại số, các biến PHP được sử dụng để giữ các giá trị hoặc biểu thức.

Một biến có thể có một tên ngắn, như x, hoặc một cái tên dài hơn, như họ và tên chúng ta vậy.

Quy tắc cho tên biến PHP :

- Các biến trong PHP bắt đầu với một kí hiệu \$, tiếp theo là tên của biến
- Tên biến phải bắt đầu với một ký tự hoặc ký tự gạch dưới
- Một tên biến chỉ có thể chứa các ký tự chữ-số và gạch dưới (A-z, 0-9, và _)
- Một tên biến không nên chứa không gian (gọi là khoảng cách ý vd : \$a b ← biến này là sai , \$ba ← thế này mới đúng)
- Tên biến là trường hợp nhạy cảm và quan trọng nên nó phân biệt cả hoa và thường (y và Y là hai biến khác nhau)

Creating (Declaring) PHP Variables

PHP không có lệnh để khai báo một biến.

Một biến được tạo ra thời điểm lần đầu tiên bạn chỉ định một giá trị cho nó: (lúc mới bắt đầu file php)

Vd về 1 biến đây

```
$siverdragon12="Hacker";
```

Sau khi thực hiện biến trên thì biến **siverdragon12** sẽ giữ giá trị là **"Hacker"**

Mẹo : Nếu bạn tạo một biến mà không để bất kỳ giá trị nào thì bạn nên để đặt giá trị đó là "null"

Vd :

```
$siverdragon12=null;
```

Nào , bây giờ ta sẽ tạo ra 1 biến có chứa ký tự (chữ từ a -> z) và 1 biến có chứa số (0 -> 9)

```
<?php  
$txt="Hello  
World!";  
$x=16;  
?>
```

Chú ý : đối với chữ (ký tự) thì bạn phải để dấu ngoặc kép như trên , còn đối với số thì không cần .

PHP là một ngôn ngữ lỏng lẻo

Trong PHP, một biến không cần phải được khai báo trước khi thêm một giá trị cho nó. Trong ví dụ trên, nhận thấy rằng chúng tôi không phải nói với PHP kiểu dữ liệu biến. PHP sẽ tự động chuyển biến cho đúng kiểu dữ liệu, tùy thuộc vào giá trị của nó. Trong một ngôn ngữ lập trình mạnh mẽ, bạn phải khai báo (xác định) các loại và tên của biến trước khi sử dụng nó.

Phạm Vi PHP

Phạm vi của một biến là một phần của kịch bản, trong đó biến có thể được tham chiếu.

PHP có bốn phạm vi biến khác nhau:

- local
- global
- static
- parameter

Phạm Vi Local

Một biến khai báo **trong** một hàm PHP là địa phương và chỉ có thể được truy cập trong phạm vi chức năng đó. (Biến có phạm vi local): Ví dụ .

```
<?php
$a = 5; // phạm vi local

function myTest()
{
    echo $a; // phạm vi local
}

myTest();
?>
```

Các kịch bản trên sẽ không sản xuất bất kỳ sản lượng vì echo tuyên bố đề cập đến biến phạm vi địa phương biến \$a, mà đã không được chỉ định một giá trị trong phạm vi này. Bạn có thể có các biến địa phương có cùng tên trong các chức năng khác nhau, bởi vì các biến địa phương chỉ được công nhận bởi các chức năng mà chúng được khai báo. Các biến địa phương sẽ bị xóa ngay sau khi chức năng, nhiệm vụ được hoàn thành.

Phạm Vi Global

Phạm vi toàn cầu đề cập đến bất kỳ biến được định nghĩa bên ngoài của bất kỳ chức năng.

Biến toàn cầu có thể được truy cập từ bất kỳ một phần của kịch bản đó không phải là bên trong một hàm.

Để truy cập vào một biến toàn cầu từ bên trong một chức năng, sử dụng **global** từ khóa:

Ví dụ :

```
<?php
$a = 5;
$b = 10;

function myTest()
{
    global $a, $b;
    $b = $a + $b;
}

myTest();
echo $b;
?>
```

Kết quả đoạn code trên là 15 (sau khi chạy)

PHP cũng lưu trữ tất cả các biến toàn cầu trong một mảng gọi là \$GLOBALS [*index*]. Chỉ số của nó là tên của biến. Mảng này cũng có thể truy cập từ bên trong chức năng và có thể được sử dụng để cập nhật các biến toàn cầu trực tiếp.

Ví dụ trên có thể được viết lại như thế này:

```
<?php
$a = 5;
$b = 10;

function myTest()
{
    $GLOBALS['b'] = $GLOBALS['a'] + $GLOBALS['b'];
}

myTest();
echo $b;
?>
```

Phạm Vi Static

Khi một hàm được hoàn thành, tất cả các biến của nó thường bị xóa. Tuy nhiên, đôi khi bạn muốn biến địa phương để không bị xóa.

Để làm điều này, sử dụng các từ khóa **tĩnh** khi lần đầu tiên bạn khai báo các biến:

```
static $rememberMe;
```

Sau đó, mỗi khi hàm được gọi, biến đó vẫn sẽ có những thông tin nó có từ thời gian qua các chức năng được gọi.

Lưu ý: biến vẫn là local đến chức năng.

Parameters

Một tham số là một biến địa phương có giá trị được truyền cho hàm mã gọi.

Các thông số được khai báo trong danh sách tham số như là một phần của việc kê khai hàm functions:

```
Function myTest($para1,$para2,...)

{
// function code
}
```

Các thông số còn được gọi là đối số. Chúng tôi sẽ thảo luận chi tiết hơn khi chúng ta nói về các chức năng.

Chapter 6 : Chuỗi Biến PHP

Chuỗi biến được sử dụng sử dụng cho các giá trị có chứa ký tự .

Trong chương trình này , chúng ta sẽ nhìn vào chức năng phổ biến nhất và vận hành được sử dụng để thao tác các chuỗi biến trong PHP.

Sau khi chúng tôi tạo ra một chuỗi , chúng ta có thể thao tác nó . Một chuỗi có thể được sử dụng trực tiếp trong một hàm hoặc nó có thể được lưu trữ trong một biến .

Dưới đây, các tập lệnh PHP gán văn bản "Hello World" vào một chuỗi biến gọi là \$ txt:

```
<?php
$txt="Hello World";
echo $txt;
?>
```

Khi thực thi code trên thì sẽ ra kết quả trả về là “Hello World”

Nào , bây giờ chúng ta thử 1 số chức năng và các chức năng khác nhau để xử lý 1 chuỗi

Chỉ có một chuỗi trong PHP.

Sử dụng dấu (.) Được sử dụng để đặt hai chuỗi với nhau.

Để nối hai biến chuỗi với nhau, sử dụng toán tử nối:

```
<?php
$txt1="Hello World!";
$txt2="What a nice day!";
echo $txt1 . " " . $txt2;
?>
```

Kết quả trả về sẽ là

Hello World ! What a nice day!

Nếu chúng ta nhìn vào đoạn mã trên, bạn thấy rằng chúng ta sử dụng toán tử nối hai lần. Điều này là bởi vì chúng tôi đã để chèn một chuỗi thứ ba (một nhân vật không gian), để phân cách hai chuỗi.

Hàm strlen()

Hàm strlen () được sử dụng để trả lại chiều dài của một chuỗi.

Hãy tìm chiều dài của một chuỗi:

```
<?php
echo strlen("Hello world!");
?>
```

Kết quả trả về là

12

Chiều dài của một chuỗi thường được sử dụng trong vòng lặp hoặc các chức năng khác, điều quan trọng là để biết khi nào kết thúc chuỗi. (tức là trong một vòng lặp, chúng tôi muốn ngừng vòng lặp sau khi ký tự cuối cùng trong chuỗi).

Hàm strpos()

Hàm strpos () được sử dụng để tìm kiếm một nhân vật / văn bản trong một chuỗi.

Nếu kết hợp được tìm thấy, chức năng này sẽ trở lại với vị trí nhân vật của trận đấu đầu

tiên. Nếu không phù hợp được tìm thấy, nó sẽ trả về FALSE.

Hãy xem nếu chúng ta có thể tìm thấy chuỗi "thế giới" trong chuỗi ký tự của chúng tôi:

```
<?php
echo strpos("Hello world!","world");
?>
```

Kết quả trả về là

6

Vị trí từ "world" trong ví dụ trên là 6. Lý do mà nó là 6 (và không phải 7), là vị trí của ký tự đầu tiên trong chuỗi là 0, và không phải 1.

Thông tin về 1 số chức năng của các hàm khác (có thể là không đầy đủ nhưng cũng nên đọc qua cho biết)

http://w3schools.com/php/php_ref_string.asp

Chapter 7 : Toán Tử PHP

Toán Tử (Đại số đơn giản) Operators

The table below lists the arithmetic operators in PHP:

Operator	Name	Description	Example	Result
$x + y$	Addition . Cộng	Sum of x and y	$2 + 2$	4
$x - y$	Subtraction . Trừ	Difference of x and y	$5 - 2$	3
$x * y$	Multiplication. Nhân	Product of x and y	$5 * 2$	10
x / y	Division . Chia	Quotient of x and y	$15 / 5$	3
$x \% y$	Modulus . Chia lấy số dư	Remainder of x divided by y	$5 \% 2$ $10 \% 8$ $10 \% 2$	1 2 0
$- x$	Negation . số âm	Opposite of x	$- 2$	
$a . b$	Concatenation . Ghép kí tự	Concatenate two strings	"Hi" . "Ha"	HiHa

Phân công Operators

The basic assignment operator in PHP is "=". It means that the left operand gets set to the value of the expression on the right. That is, the value of "\$x = 5" is 5.

Assignment Same as...		Description
x = y	x = y	The left operand gets set to the value of the expression on the right
x += y	x = x + y	Addition
x -= y	x = x - y	Subtraction
x *= y	x = x * y	Multiplication
x /= y	x = x / y	Division
x %= y	x = x % y	Modulus
a .= b	a = a . b	Concatenate two strings

Tăng dần/giảm dần Operators

Operator	Name	Description
++ x	Pre-increment	Increments x by one, then returns x
x ++	Post-increment	Returns x, then increments x by one
-- x	Pre-decrement	Decrements x by one, then returns x
x --	Post-decrement	Returns x, then decrements x by one

Toán Tử So Sánh

Comparison operators allows you to compare two values:

Operator	Name	Description	Example
x == y	Equal	True if x is equal to y	5==8 returns false
x === y	Identical	True if x is equal to y, and they are of same type	5==="5" returns false
x != y	Not equal	True if x is not equal to y	5!=8 returns true
x <> y	Not equal	True if x is not equal to y	5<>8 returns true
x !== y	Not identical	True if x is not equal to y, or they are not of same type	5!== "5" returns true
x > y	Greater than	True if x is greater than y	5>8 returns false
x < y	Less than	True if x is less than y	5<8 returns true
x >= y	Greater than or equal to	True if x is greater than or equal to y	5>=8 returns false
x <= y	Less than or	True if x is less than or equal to	5<=8 returns true

equal to y

Logical Operators

Operator	Name	Description	Example
x and y	And	True if both x and y are true	x=6 y=3 (x < 10 and y > 1) returns true
x or y	Or	True if either or both x and y are true	x=6 y=3 (x==6 or y==5) returns true
x xor y	Xor	True if either x or y is true, but not both	x=6 y=3 (x==6 xor y==3) returns false
x && y	And	True if both x and y are true	x=6 y=3 (x < 10 && y > 1) returns true
x y	Or	True if either or both x and y are true	x=6 y=3 (x==5 y==5) returns false
! x	Not	True if x is not true	x=6 y=3 !(x==y) returns true

Array Operators

Operator	Name	Description
x + y	Union	Union of x and y
x == y	Equality	True if x and y have the same key/value pairs
x === y	Identity	True if x and y have the same key/value pairs in the same order and of the same types
x != y	Inequality	True if x is not equal to y
x <> y	Inequality	True if x is not equal to y
x !== y	Non-identity	True if x is not identical to y

Chapter 8 : Câu lệnh If ... else trong PHP

Câu điều kiện được sử dụng để thực hiện các hành động khác nhau dựa trên các điều kiện khác nhau.

Điều kiện câu lệnh (đọc cho biết và suy nghĩ) .

Nó rất được dùng thường xuyên khi bạn viết code , bạn muốn thực hiện một hành động (1 câu lệnh) nhưng với những quyết định khác nhau thì câu lệnh if ... else là sự lựa chọn của bạn

Bạn có thể sử dụng câu lệnh điều kiện trong mã của bạn để làm điều này.

- Trong PHP chúng tôi có báo cáo điều kiện sau đây .
- Câu lệnh if (không có else) – cái này dùng để thực hiện một lệnh chỉ khi điều kiện quy định là đúng sự thật
- Câu lệnh if ... else – thực hiện điều kiện nếu nó điều kiện là đúng còn nếu sai thì sẽ trả kết quả khác ở else
- Câu lệnh if ... elseif ... else – dùng để thực hiện nhiều khối mã 1 lúc trong nhiều khối mã đã thực hiện (vd ở trên có mỗi 1 điều kiện sai thì trả về cái kia , còn cái này nhiều trường hợp đại loại thế)
- Câu lệnh switch – Sử dụng câu lệnh này để lựa chọn một trong nhiều đoạn mã sẽ được thực hiện .

Câu lệnh IF

- Sử dụng câu lệnh if khi thực hiện một số mã khi và chỉ khi điều kiện qui định của nó đúng với mã được qui định

Cú pháp câu if :

If (điều kiện) sẽ được thực hiện nếu điều kiện đúng .
--

Ví dụ sau đây sẽ đưa ra "Have a nice weekend!" nếu ngày hiện nay là thứ Sáu:

```
<html>
<body>

<?php
$d=date("D");
if ($d=="Fri") echo "Have a nice weekend!";
?>

</body>
</html>
```

Chú ý rằng đoạn mã trên không có else . vì vậy mã này được thực thi khi nó chỉ đúng với điều kiện .

Câu lệnh IF ... ELSE

Sử dụng câu lệnh if ... else để thực hiện đoạn mã nếu điều kiện nó là đúng hoặc sai .

Cú pháp :

```
if (điều kiện)
{
    Mã lệnh sẽ được thực hiện nếu điều kiện đúng;
}
else
{
    Mã lệnh sẽ được thực hiện nếu điều kiện sai;
}
```

Ví dụ

Ví dụ sau đây sẽ đưa ra "Have a nice weekend!" nếu ngày hiện nay là thứ Sáu, nếu không phải nó sẽ trả kết quả khác là "Have a nice day!"

```
<html>
<body>

<?php
$d=date("D");
if ($d=="Fri")
{
    echo "Have a nice weekend!";
}
else
{
    echo "Have a nice day!";
}
?>

</body>
</html>
```

Câu lệnh IF ... ELSEIF ... ELSE

Sử dụng câu lệnh if elseif ... else để lựa chọn một trong nhiều khối mã sẽ được thực hiện.

Cú pháp

```
if (Điều kiện 1)
{
    code sẽ được thực hiện nếu điều kiện 1 đúng;
}
elseif (Điều kiện 2)
{
    code sẽ được thực hiện nếu điều kiện 2 đúng;
}
else
{
    code sẽ được thực hiện nếu mấy cái điều kiện trên sai;
}
```

Ví dụ

Ví dụ sau đây sẽ đưa ra "Have a nice weekend!" nếu ngày hiện nay là thứ Sáu, và "Have a nice Sunday!" nếu ngày hiện nay là chủ nhật. Nếu không, nó sẽ ra "Have a nice day!"

```
<html>
<body>

<?php
$d=date("D");
if ($d=="Fri")
{
    echo "Have a nice weekend!";
}
elseif ($d=="Sun")
{
    echo "Have a nice Sunday!";
}
else
{
    echo "Have a nice day!";
}
?>

</body>
</html>
```

Chapter 9 : Câu lệnh Switch PHP

Câu điều kiện được sử dụng để thực hiện các hành động khác nhau dựa trên các điều kiện khác nhau.

Sử dụng câu lệnh switch để lựa chọn một trong nhiều khối mã sẽ được thực hiện.

Cú pháp

```
switch (n)
{
case label1:
    code sẽ thực hiện nếu n=label1;
    break;
case label2:
    code sẽ thực hiện nếu if n=label2;
    break;
default:
    code sẽ thực hiện nếu n khác cả 2 cái label1 và label2;
}
```

Đây là cách nó hoạt động: Đầu tiên chúng ta có một biểu thức duy nhất n (thường xuyên nhất một biến), được đánh giá một lần. Giá trị của biểu thức sau đó được so sánh với các giá trị cho từng trường hợp trong cấu trúc. Nếu có một trận đấu, các khối mã liên kết với trường hợp đó được thực hiện. Sử dụng `break` để ngăn chặn các mã từ chạy vào trường hợp tiếp theo tự động. Những tuyên bố mặc định được sử dụng nếu không tìm thấy.

Ví dụ

```
<html>
<body>

<?php
$x=1;
switch ($x)
{
case 1:
    echo "Number 1";
    break;
case 2:
    echo "Number 2";
    break;
case 3:
    echo "Number 3";
    break;
default:
    echo "No number between 1 and 3";
}
?>

</body>
</html>
```

Chapter 10 : Arrays (Mảng)

Một mảng lưu trữ nhiều giá trị trong một biến duy nhất.

Một Array là gì?

Một biến là một vùng lưu trữ đang nắm giữ một số hoặc văn bản. Vấn đề là, một biến sẽ chỉ có một giá trị.

Một mảng là một biến đặc biệt, có thể lưu trữ nhiều giá trị trong một biến duy nhất.

Nếu bạn có một danh sách các mục (một danh sách các tên xe, ví dụ), lưu trữ các xe ô tô trong các biến duy nhất có thể nhìn như thế này:

```
$cars1="Saab";
$cars2="Volvo";
$cars3="BMW";
```

Tuy nhiên , nếu không phải 3 cái tên mà là hàng ngàn cái tên thì sao ? Giải pháp tốt nhất đó là dùng mảng

Một mảng có thể giữ tất cả các giá trị biến của bạn dưới một cái tên. Và bạn có thể truy cập vào các giá trị bằng cách đề cập đến tên mảng.

Mỗi phần tử trong mảng có chỉ số riêng của mình để nó có thể dễ dàng truy cập.

Trong PHP, có ba loại của các mảng:

Numeric array - Một mảng với một số chỉ số

Associative array - Một mảng mà mỗi phím ID được kết hợp với một giá trị

Multidimensional array- Một mảng có chứa một hoặc nhiều mảng

Numeric Arrays

Một mảng số các cửa hàng mỗi phần tử mảng với một số chỉ số.

Có hai phương pháp để tạo ra một mảng số.

1. Trong ví dụ sau chỉ số được gán tự động (chỉ số bắt đầu từ 0):

```
$cars=array("Saab","Volvo","BMW","Toyota");
```

2. Trong ví dụ sau chúng ta gán các chỉ số theo cách thủ công:

```
$cars[0]="Saab";  
$cars[1]="Volvo";  
$cars[2]="BMW";  
$cars[3]="Toyota";
```

Ví dụ

Trong ví dụ sau bạn truy cập các giá trị biến bằng cách đề cập đến tên mảng và chỉ số:

```
<?php
$cars[0]="Saab";
$cars[1]="Volvo";
$cars[2]="BMW";
$cars[3]="Toyota";
echo $cars[0] . " and " . $cars[1] . " are Swedish cars.";
?>
```

Code sẽ trả về là

```
Saab and Volvo are Swedish cars.
```

Associative Arrays

Một mảng kết hợp, mỗi phím ID được kết hợp với một giá trị.

Khi lưu trữ dữ liệu về các giá trị cụ thể được đặt tên, một mảng số không phải là luôn luôn là cách tốt nhất để làm điều đó.

Với mảng kết hợp, chúng tôi có thể sử dụng các giá trị như là chìa khóa và gán giá trị cho họ.

Ví dụ 1

Trong ví dụ này, chúng tôi sử dụng một mảng để gán lứa tuổi để người khác nhau:

```
$ages = array("Peter"=>32, "Quagmire"=>30, "Joe"=>34);
```

Ví dụ 2

Ví dụ này là tương tự như ví dụ 1, nhưng cho thấy một cách khác nhau của việc tạo ra các mảng:

```
$ages['Peter'] = "32";
$ages['Quagmire'] = "30";
$ages['Joe'] = "34";
```

Các phím ID có thể được sử dụng trong một kịch bản:

```
<?php
$ages['Peter'] = "32";
$ages['Quagmire'] = "30";
$ages['Joe'] = "34";

echo "Peter is " . $ages['Peter'] . " years old.";
?>
```

Code sẽ trả về là

```
Peter is 32 years old.
```

Multidimensional Arrays

Trong một mảng đa chiều, mỗi phần tử trong mảng chính cũng có thể là một mảng. Và mỗi phần tử trong mảng phụ có thể là một mảng, và như vậy.

Ví dụ

Trong ví dụ này, chúng ta tạo một mảng đa chiều, với các phím ID tự động gán:

```
$families = array
(
  "Griffin"=>array
  (
    "Peter",
    "Lois",
    "Megan"
  ),
  "Quagmire"=>array
  (
    "Glenn"
  ),
  "Brown"=>array
  (
    "Cleveland",
    "Loretta",
    "Junior"
  )
);
```

Các mảng ở trên sẽ giống như thế này nếu bằng văn bản đến đầu ra:

```
Array
(
    [Griffin] => Array
        (
            [0] => Peter
            [1] => Lois
            [2] => Megan
        )
    [Quagmire] => Array
        (
            [0] => Glenn
        )
    [Brown] => Array
        (
            [0] => Cleveland
            [1] => Loretta
            [2] => Junior
        )
)
```

Ví dụ 2

Cho phép thử hiển thị một giá trị từ mảng ở trên:

```
echo "Is " . $families['Griffin'][2] .  
" a part of the Griffin family?";
```

Kết quả trả về sẽ là

```
Is Megan a part of the Griffin family?
```

Chapter 11 : Looping – While Loops

Loops thực thi một khối mã một số quy định của lần, hoặc trong khi một điều kiện được chỉ định là đúng sự thật.

PHP Loops

Thông thường khi bạn viết mã, bạn muốn cùng một khối mã để chạy hơn và hơn nữa trong một hàng. Thay vì thêm một vài dòng trong một kịch bản gần như bằng nhau, chúng ta có thể sử dụng các vòng lặp để thực hiện một nhiệm vụ như thế này.

Trong PHP, chúng tôi có báo cáo vòng lặp sau đây:

- while - loops thông qua một khối mã trong khi một điều kiện được chỉ định là đúng sự thật
 - do ... while - vòng lặp thông qua một khối mã một lần, và sau đó lặp đi lặp lại vòng lặp dài như một điều kiện xác định là đúng sự thật
 - foreach - loops một khối mã cho mỗi phần tử trong một mảng
-

The while Loop

Các vòng lặp trong khi thực thi một khối mã trong khi một điều kiện là đúng sự thật.

Cú pháp

```
while (Điều kiện)  
{  
    code sẽ được thực hiện;  
}
```

Ví dụ

Ví dụ dưới đây đầu tiên thiết lập một biến 1 (\$i=1;).

Sau đó, trong khi vòng lặp sẽ tiếp tục chạy khi i nhỏ hơn hoặc bằng 5. Nó sẽ tăng thêm 1 mỗi lần vòng lặp chạy:

```
<html>  
<body>  
  
<?php  
$i=1;  
while($i<=5)  
{  
    echo "The number is " . $i . "<br>";  
    $i++;  
}  
?>  
  
</body>  
</html>
```

Kết quả là

```
The number is 1
The number is 2
The number is 3
The number is 4
The number is 5
```

The do...while Statement

Việc do ... while khi tuyên bố sẽ luôn luôn thực hiện các khối mã một lần, sau đó sẽ kiểm tra các điều kiện, và lặp lại vòng lặp trong khi điều kiện là đúng.

Cú pháp

```
do
{
    code sẽ được thực hiện;
}
while (Điều kiện);
```

Ví dụ

Ví dụ dưới đây đầu tiên thiết lập một biến 1 ($i=1$;

Sau đó, nó bắt đầu do ... trong khi vòng lặp. Các vòng lặp sẽ tăng biến i với 1, và sau đó viết một số đầu ra. Sau đó, điều kiện được kiểm tra (là i ít hơn, hoặc bằng 5), và vòng lặp sẽ tiếp tục chạy khi i nhỏ hơn hoặc bằng 5

```
<html>
<body>

<?php
$i=1;
do
{
    $i++;
    echo "The number is " . $i . "<br>";
}
while ($i<=5);
?>

</body>
</html>
```

Kết quả cho ra là

```
The number is 2
The number is 3
The number is 4
The number is 5
The number is 6
```

Các vòng lặp và vòng lặp foreach sẽ được giải thích trong chương kế tiếp.

Chapter 12 : Looping – For Loops

Loops thực thi một khối mã một số quy định của lần, hoặc trong khi một điều kiện được chỉ định là đúng sự thật.

The for Loop

Vòng lặp for được sử dụng khi bạn biết trước kịch bản nên chạy bao nhiêu lần.
Cú pháp

```
for (init; condition; increment)  
{  
    code to be executed;  
}
```

Chú thích:

init: Hầu hết sử dụng số đếm (nhưng có thể có bất kỳ mã được thực hiện một lần vào đầu của vòng lặp)

condition: Điều kiện cho mỗi lần lặp. Nếu điều kiện này đúng, vòng lặp tiếp tục. Nếu nó điều kiện sai thì vòng lặp kết thúc.

increment: Hầu hết để tăng số đếm (nhưng có thể có bất kỳ mã sẽ được thực hiện vào cuối của sự lặp lại)

Lưu ý: Các thông số **init** và **increment** có thể được sản phẩm nào hoặc có nhiều biểu ngữ (cách nhau bằng dấu phẩy).

Ví dụ

Ví dụ dưới đây định nghĩa một vòng lặp bắt đầu với $i = 1$. Các vòng lặp sẽ tiếp tục chạy miễn là biến i nhỏ hơn hoặc bằng 5. Biến i sẽ tăng thêm 1 mỗi lần vòng lặp chạy:

```
<html>
<body>

<?php
for ($i=1; $i<=5; $i++)
{
    echo "The number is " . $i . "<br>";
}
?>

</body>
</html>
```

Kết quả trả về là

The number is 1
The number is 2
The number is 3
The number is 4
The number is 5

The foreach Loop

Vòng lặp foreach được sử dụng để lặp qua mảng.
Cú pháp

```
foreach ($array as $value)
{
    code to be executed;
}
```

Đối với mỗi vòng lặp, giá trị của phần tử mảng hiện tại được gán cho \$ value (và con trỏ mảng là di chuyển của một) - do đó, vòng lặp tiếp theo, bạn sẽ được xem xét giá trị mảng tiếp theo.

Ví dụ

Ví dụ sau đây cho thấy một vòng lặp mà sẽ in các giá trị của các mảng nhất định

```
<html>
<body>

<?php
$x=array("one","two","three");
foreach ($x as $value)
{
    echo $value . "<br>";
}
?>

</body>
</html>
```

Kết quả trả về sẽ là

```
one
two
three
```

Chapter 13 : Functions (Hàm Chức Năng)

Sức mạnh thực sự của PHP xuất phát từ chức năng của mình.

Trong PHP, có hơn 700 tích hợp chức năng.

Bạn có thể tham khảo các chức năng nhiều hơn ở đây

<http://w3schools.com/php/default.asp>

PHP Functions

Trong chương này, chúng tôi sẽ cho bạn thấy làm thế nào để tạo ra các chức năng của riêng bạn. Để giữ cho kịch bản đang được thực thi khi tải trang, bạn có thể đặt nó vào một hàm. Chức năng sẽ được thực hiện bởi một cuộc gọi đến chức năng. Bạn có thể gọi một chức năng từ bất cứ nơi nào trong một trang .

Create a PHP Function

Chức năng sẽ được thực hiện bởi một cuộc gọi đến chức năng.

Cú pháp

```
function functionName()  
{  
    code to be executed;  
}
```

PHP function hướng dẫn:

Cung cấp cho các chức năng một tên phản ánh những gì chức năng
Tên chức năng có thể bắt đầu bằng một chữ cái hoặc gạch dưới (không phải số)

Ví dụ

Một chức năng đơn giản, viết tên của tôi khi nó được gọi là:

```
<html>  
<body>  
  
<?php  
function writeName()  
{  
    echo "siverdragon12";  
}  
  
echo "My name is ";  
writeName();  
?>  
  
</body>  
</html>
```

Kết quả là

```
My name is siverdragon12
```

PHP Functions - Adding parameters (thêm các thông số)

Để thêm nhiều chức năng cho một hàm, chúng ta có thể thêm các thông số. Một tham số là giống như một biến.

Tham số được quy định cụ thể sau tên hàm, bên trong dấu ngoặc đơn.

Ví dụ 1

Ví dụ sau đây sẽ viết tên khác nhau đầu tiên, nhưng tên cuối cùng như nhau:

```
<html>
<body>

<?php
function writeName($fname)
{
echo $fname . "
Refsnes.<br>";
}

echo "My name is ";
writeName("Kai Jim");
echo "My sister's name is ";
writeName("Hege");
echo "My brother's name is ";
writeName("Stale");
?>

</body>
</html>
```

Kết quả sẽ là

```
My name is Kai Jim Refsnes.
My sister's name is Hege Refsnes.
My brother's name is Stale Refsnes.
```

Ví dụ 2

Các chức năng sau đây có hai tham số:

```
<html>
<body>

<?php
function writeName($fname,$punctuation)
{
echo $fname . " Refsnes" . $punctuation . "<br>";
}

echo "My name is ";
writeName("Kai Jim",".");
echo "My sister's name is ";
writeName("Hege","!");
echo "My brother's name is ";
writeName("Ståle","?");
?>

</body>
</html>
```

Kết quả sẽ là

```
My name is Kai Jim Refsnes.
My sister's name is Hege Refsnes!
My brother's name is Ståle Refsnes?
```

PHP Functions - Return values

Để cho phép một hàm trả về một giá trị, sử dụng các tuyên bố trở lại.

Ví dụ

```
<html>
<body>

<?php
function add($x,$y)
{
$total=$x+$y;
return $total;
}

echo "1 + 16 = " . add(1,16);
?>

</body>
</html>
```

Kết quả sẽ là

1 + 16 = 17

Chapter 14 : PHP Forms and User Input

Các biến PHP `$_GET` và `$_POST` được sử dụng để lấy thông tin từ các hình thức, như đầu vào của người sử dụng.

PHP Form Handling

Điều quan trọng nhất để ý khi giao dịch với các hình thức HTML và PHP là bất kỳ yếu tố hình thức trong một trang HTML sẽ tự động có sẵn cho PHP script của bạn.

Ví dụ

Ví dụ dưới đây có chứa một dạng HTML với hai lĩnh vực đầu vào và một nút submit:

```
<html>
<body>

<form action="welcome.php" method="post">
Name: <input type="text" name="fname">
Age: <input type="text" name="age">
<input type="submit">
</form>

</body>
</html>
```

Khi một người dùng điền vào các hình thức trên và nhấp chuột vào nút submit, các dữ liệu mẫu được gửi vào một tập tin PHP, gọi là "welcome.php":

"welcome.php" trông như thế này:

```
<html>
<body>

Welcome <?php echo $_POST["fname"]; ?>!<br>
You are <?php echo $_POST["age"]; ?> years old.

</body>
</html>
```

Kết quả trả về sẽ đại loại như thế này

```
Welcome John!
You are 28 years old.
```

Các biến PHP \$_POST và \$_GET sẽ được giải thích trong các chương tiếp theo.

Form Validation

Đầu vào sử dụng nên được xác nhận trên các trình duyệt bất cứ khi nào có thể (theo lệnh của khách hàng). Trình duyệt xác nhận là nhanh hơn và giảm tải máy chủ.

Bạn nên xem xét máy chủ xác nhận nếu người dùng nhập vào sẽ được chèn vào cơ sở dữ liệu. Một cách tốt để xác nhận một hình thức trên máy chủ là để viết các hình thức cho chính nó, thay vì nhảy đến một trang khác. Sau đó người dùng sẽ nhận được các thông

báo lỗi trên cùng một trang như hình thức. Điều này làm cho nó dễ dàng hơn để phát hiện ra các lỗi.

Chapter 15 : PHP \$_GET Variable

Trong PHP, biến \$_GET được xác định trước được sử dụng để thu thập các giá trị trong một hình thức với phương pháp = "get".

The \$_GET Variable

Thông tin được gửi từ một hình thức với phương thức GET có thể nhìn thấy tất cả mọi người (nó sẽ được hiển thị trong thanh địa chỉ của trình duyệt) và có giới hạn về số lượng thông tin để gửi.

Ví dụ

```
<form action="welcome.php" method="get">
Name: <input type="text" name="fname">
Age: <input type="text" name="age">
<input type="submit">
</form>
```

Khi người dùng nhấp vào nút "Submit", URL gửi đến máy chủ có thể nhìn một cái gì đó như thế này:

```
http://localhost/welcome.php?fname=Peter&age=37
```

Các tập tin "welcome.php" bây giờ có thể sử dụng biến \$_GET để thu thập dữ liệu mẫu (tên của các trường mẫu tự động sẽ là chìa khóa trong mảng \$_GET):

```
Welcome <?php echo $_GET["fname"]; ?>.<br>
You are <?php echo $_GET["age"]; ?> years old!
```

When to use method="get"?

Khi sử dụng phương pháp "get" trong các hình thức HTML, tất cả các tên biến và giá trị được hiển thị trong URL.

Lưu ý: Phương pháp này không nên được sử dụng khi gửi mật khẩu hoặc các thông tin nhạy cảm khác!

Tuy nhiên, bởi vì các biến được hiển thị trong URL, nó có thể đánh dấu trang. Điều này có thể hữu ích trong một số trường hợp.

Lưu ý: Phương pháp get là không phù hợp với các giá trị biến rất lớn. Nó không nên được sử dụng với giá trị vượt quá 2000 ký tự.

Chapter 16 : PHP \$_POST Function

Trong PHP, biến \$_POST được xác định trước được sử dụng để thu thập các giá trị trong một hình thức với phương pháp = "post".

The \$_POST Variable (Biến \$_POST)

Biến \$_POST được xác định trước được sử dụng để thu thập các giá trị từ một hình thức gửi với phương pháp = "post".

Thông tin được gửi từ một hình thức với phương thức POST là vô hình cho người khác và không có giới hạn về số lượng thông tin để gửi.

Lưu ý: Tuy nhiên, kích thước tối đa cho phương thức POST là 8 MB mặc định (có thể thay đổi mặc định thông qua file php.ini , dòng post_max_size)

Ví dụ :

```
<form action="welcome.php" method="post">
Name: <input type="text" name="fname">
Age: <input type="text" name="age">
<input type="submit">
</form>
```

Khi người dùng nhấp vào nút "Submit", URL sẽ như thế này:

<http://localhost/welcome.php>

Các tập tin "welcome.php" bây giờ có thể sử dụng biến \$ _POST để thu thập dữ liệu mẫu (tên của các trường mẫu tự động sẽ là chìa khóa trong mảng \$ _POST):

```
Welcome <?php echo $ _POST["fname"]; ?>!  
You are <?php echo $ _POST["age"]; ?> years old.
```

When to use method="post"?

Thông tin được gửi từ một hình thức với phương thức POST là vô hình cho người khác và không có giới hạn về số lượng thông tin để gửi.

Tuy nhiên, bởi vì các biến không được hiển thị trong URL, nó không phải là có thể đánh dấu trang.

The PHP \$ _REQUEST Variable

Biến \$ _REQUEST được xác định trước có chứa các nội dung của cả hai \$ _GET, \$ _POST , \$ _COOKIE

Biến \$ _REQUEST có thể được sử dụng để thu thập dữ liệu mẫu được gửi với cả hai phương pháp GET và POST.

Ví dụ

```
Welcome <?php echo $ _REQUEST["fname"]; ?>!  
You are <?php echo $ _REQUEST["age"]; ?> years old.
```

Phần 2 : PHP Nâng Cao

Chapter 17 : Date() Function

Chức năng php date() dùng để định dạng một thời gian và / hoặc ngày.

The PHP Date() Function

Chức năng PHP Date() định dạng dấu thời gian cho một ngày dễ đọc hơn và thời gian.

Mẹo : Dấu thời gian là một chuỗi các ký tự, biểu thị ngày tháng và / hoặc thời gian mà tại đó một sự kiện nào đó xảy ra.

Cú pháp

`date(format,timestamp)`

Parameter	Description
format	Required. Specifies the format of the timestamp
timestamp	Optional. Specifies a timestamp. Default is the current date and time

PHP Date() - Format the Date

Tham số định dạng cần thiết trong date() xác định làm thế nào để định dạng ngày / thời gian.

Dưới đây là một số ký tự có thể được sử dụng:

- d - Đại diện các ngày trong tháng (01-31)
- m - Đại diện cho một tháng (01-12)
- Y - Đại diện một năm (bốn chữ số)

Bạn có thể tham khảo thêm tại đây

http://w3schools.com/php/php_ref_date.asp

Những ký tự khác như : “/”, “.”, “-” cũng có thể được chèn vào giữa các các chữ cái để bổ sung thêm định dạng .

```
<?php
echo date("Y/m/d") . "<br>";
echo date("Y.m.d") . "<br>";
echo date("Y-m-d");
?>
```

Kết quả trả về sẽ là

```
2013/05/11
2013.05.11
2013-05-11
```

PHP Date() - Adding a Timestamp

Tham số dấu thời gian tùy chọn trong date() chức năng xác định một dấu thời gian. Nếu bạn không chỉ định một dấu thời gian, ngày hiện tại và thời gian sẽ được sử dụng.

Chức năng mktime() trả về các dấu thời gian Unix cho một ngày.

Các dấu thời gian Unix chứa số giây giữa Epoch Unix (January 1 1970 00:00:00 GMT) và thời gian quy định.

Cú pháp cho mktime()

```
mktime(hour,minute,second,month,day,year,is_dst)
```

Để đi một ngày trong tương lai, chúng tôi chỉ đơn giản là thêm một đối số ngày mktime()

```
<?php
$tomorrow = mktime(0,0,0,date("m"),date("d")+1,date("Y"));
echo "Tomorrow is ".date("Y/m/d", $tomorrow);
?>
```

Code sẽ ra là

```
Tomorrow is 2009/05/12
```

Chapter 17 : Include File

PHP include and require Statements

Trong PHP, bạn có thể chèn nội dung của một file PHP vào một tập tin PHP trước khi máy chủ thực hiện nó.

Bao gồm và yêu cầu báo cáo được sử dụng để chèn mã hữu ích bằng văn bản trong các tập tin khác, trong dòng chảy của thực hiện.

Bao gồm và yêu cầu giống hệt nhau, ngoại trừ khi thất bại:

- yêu cầu sẽ tạo ra một lỗi nghiêm trọng (E_COMPILE_ERROR) và dừng lại kịch bản
- bao gồm sẽ chỉ sản xuất một cảnh báo (E_WARNING) và kịch bản sẽ tiếp tục

Vì vậy, nếu bạn muốn thực hiện để đi vào và chỉ cho người sử dụng đầu ra, ngay cả khi bao gồm các tập tin là mất tích, sử dụng bao gồm. Nếu không, trong trường hợp FrameWork, CMS hoặc một ứng dụng PHP mã hóa phức tạp, luôn luôn sử dụng yêu cầu để bao gồm một tập tin quan trọng đến dòng chảy của thực hiện. Điều này sẽ giúp tránh ảnh hưởng đến an ninh và toàn vẹn của ứng dụng của bạn, chỉ cần trong trường hợp một trong những tập tin quan trọng là vô tình mất tích.

Các file bao gồm tiết kiệm rất nhiều công việc. Điều này có nghĩa là bạn có thể tạo một tiêu đề tiêu chuẩn, chân, hoặc tập tin trình đơn cho tất cả các trang web của bạn. Sau đó, khi tiêu đề cần được cập nhật, bạn chỉ có thể cập nhật các tiêu đề bao gồm các tập tin.

Syntax (Cú pháp)

```
include 'filename';
```

or

```
require 'filename';
```

PHP include and require Statement

Giả sử bạn có một tập tin tiêu đề tiêu chuẩn, được gọi là "header.php". Để bao gồm các tập tin tiêu đề trong một trang, sử dụng include/require:

```
<html>
<body>

<?php include 'header.php'; ?>
<h1>Welcome to my home page!</h1>
<p>Some text.</p>

</body>
</html>
```

Ví dụ 2

Giả sử chúng ta có một tập tin trình đơn tiêu chuẩn nên được sử dụng trên tất cả các trang.

"menu.php":

```
echo '<a href="/default.php">Home</a>
<a href="/tutorials.php">Tutorials</a>
<a href="/references.php">References</a>
<a href="/examples.php">Examples</a>
<a href="/about.php">About Us</a>
<a href="/contact.php">Contact Us</a>';
```

Tất cả các trang trong trang Web nên bao gồm các tập tin trình đơn này. Ở đây là làm thế nào nó có thể được thực hiện:

```
<html>
<body>

<div class="leftmenu">
<?php include 'menu.php'; ?>
</div>

<h1>Welcome to my home page.</h1>
<p>Some text.</p>

</body>
</html>
```

Ví dụ 3

Giả sử chúng ta có một tập tin với một số biến được định nghĩa ("vars.php") bao gồm:

```
<?php
$color='red';
$car='BMW';
?>
```

Sau đó, các biến có thể được sử dụng trong các tập tin gọi:

```
<html>
<body>

<h1>Welcome to my home page.</h1>
<?php include 'vars.php';
echo "I have a $color $car"; // I have a red BMW
?>

</body>
</html>
```

Chapter 18 : File Handling

Chức năng fopen() dùng để mở 1 file trong PHP

Opening a File

Tham số đầu tiên của chức năng này chứa tên của tập tin được mở và tham số thứ hai quy định cụ thể trong chế độ tập tin phải được mở ra:

```
<html>
<body>

<?php
$file=fopen("welcome.txt","r");
?>

</body>
</html>
```

Tập tin có thể được mở trong một trong các phương thức sau đây:

Modes	Description
r	Read only. Starts at the beginning of the file
r+	Read/Write. Starts at the beginning of the file
w	Write only. Opens and clears the contents of file; or creates a new file if it doesn't exist
w+	Read/Write. Opens and clears the contents of file; or creates a new file if it doesn't exist
a	Append. Opens and writes to the end of the file or creates a new file if it doesn't exist
a+	Read/Append. Preserves file content by writing to the end of the file
x	Write only. Creates a new file. Returns FALSE and an error if file already exists
x+	Read/Write. Creates a new file. Returns FALSE and an error if file already exists

Lưu ý: Nếu hàm fopen() là không thể mở các tập tin được chỉ định, nó sẽ trả về 0 (false).
Ví dụ

Ví dụ sau đây tạo ra một tin nhắn nếu hàm fopen() là không thể mở các tập tin được chỉ định:

```
<html>
<body>

<?php
$file=fopen("welcome.txt","r") or exit("Unable to open file!");
?>

</body>
</html>
```

Closing a File

Chức năng fclose() dùng để đóng file đang mở:

```
<?php
$file = fopen("test.txt","r");

//some code to be executed

fclose($file);
?>
```

Check End-of-file

Feof () chức năng kiểm tra nếu "end-of-file (EOF) đã đạt được.

Feof () chức năng hữu ích cho Looping thông qua các dữ liệu không rõ chiều dài.

Lưu ý: Bạn không thể đọc từ tập tin đã mở trong w, a, và chế độ x!

```
if (feof($file)) echo "End of file";
```

Reading a File Line by Line

Các fgets() chức năng được sử dụng để đọc một dòng từ một tập tin duy nhất.

Lưu ý: Sau khi một cuộc gọi đến chức năng này, con trỏ tập tin đã di chuyển đến dòng kế tiếp.

Ví dụ

Ví dụ dưới đây đọc một dòng tập tin theo dòng, cho đến khi đạt đến cuối của tập tin được:

```
<?php
$file = fopen("welcome.txt", "r") or exit("Unable to open file!");
//Output a line of the file until the end is reached
while(!feof($file))
{
    echo fgets($file). "<br>";
}
fclose($file);
?>
```

Reading a File Character by Character

Hàm `fgetc()` được sử dụng để đọc một ký tự từ một tập tin duy nhất.

Lưu ý: Sau khi một cuộc gọi đến chức năng này di chuyển con trỏ tập tin ký tự tiếp theo.
Ví dụ

Ví dụ dưới đây đọc một tập tin ký tự của nhân vật, cho đến khi đạt đến cuối tập tin:

```
<?php
$file=fopen("welcome.txt","r") or exit("Unable to open file!");
while (!feof($file))
{
    echo fgetc($file);
}
fclose($file);
?>
```

Chapter 19 : File Upload

Với PHP , nó có thể cho phép bạn upload 1 tập tin lên server (máy chủ)

Create an Upload-File Form

Để cho phép người dùng tải lên các tập tin từ một hình thức có thể rất hữu ích.

Nhìn vào các hình thức HTML cho các tập tin tải lên sau:

```
<html>
<body>

<form action="upload_file.php" method="post"
enctype="multipart/form-data">
<label for="file">Filename:</label>
<input type="file" name="file" id="file"><br>
<input type="submit" name="submit" value="Submit">
</form>

</body>
</html>
```

Chú ý sau đây về dạng HTML trên:

- Thuộc tính enctype của thẻ <form> xác định loại nội dung để sử dụng khi nộp mẫu đơn. "multipart / form-data" được sử dụng khi yêu cầu dữ liệu nhị phân, giống như nội dung của một tập tin được tải lên một hình thức (tải nhiều file cho nhanh)

- Type = "file" thuộc tính của thẻ <input> xác định rằng đầu vào nên được xử lý như một tập tin. Ví dụ, khi xem trong một trình duyệt, có sẽ là một nút browse bên cạnh lĩnh vực đầu vào

Lưu ý: Cho phép người dùng tải lên các tập tin là một nguy cơ bảo mật lớn. Chỉ cho phép người dùng tin cậy để thực hiện các tập tin tải lên.

Create The Upload Script

Nội dung file "upload_file.php" sẽ có dạng như thế này

```
<?php
if ($_FILES["file"]["error"] > 0)
{
    echo "Error: " . $_FILES["file"]["error"] . "<br>";
}
else
{
    echo "Upload: " . $_FILES["file"]["name"] . "<br>";
    echo "Type: " . $_FILES["file"]["type"] . "<br>";
    echo "Size: " . ($_FILES["file"]["size"] / 1024) . " kB<br>";
    echo "Stored in: " . $_FILES["file"]["tmp_name"];
}
?>
```

Bằng cách sử dụng PHP global \$_FILES mảng bạn có thể tải lên các tập tin từ một máy tính khách hàng đến máy chủ từ xa.

Tham số đầu tiên là tên của hình thức đầu vào và chỉ số thứ hai có thể là một trong hai "name", "type", "size", "tmp_name" hoặc "error". Như thế này:

- \$_FILES ["File"] ["name"] - tên của các tập tin tải lên
- \$_FILES ["File"] ["type"] - loại tập tin tải lên
- \$_FILES ["File"] ["size"] - kích thước trong kilobytes của các tập tin tải lên
- \$_FILES ["File"] ["tmp_name"] - tên của bản sao tạm thời của tập tin được lưu trữ trên máy chủ
- \$_FILES ["File"] ["error"] - mã lỗi từ tập tin tải lên

Đây là một cách rất đơn giản các tập tin tải lên. Vì lý do bảo mật, bạn nên thêm hạn chế về những gì người dùng được phép tải lên.

Restrictions on Upload

Trong kịch bản này, chúng ta thêm một số hạn chế để các tập tin tải lên. Người sử dụng có thể tải lên gif, jpeg, và png tập tin và kích thước tập tin phải dưới 20 kB.:

```

<?php
$allowedExts = array("jpg", "jpeg", "gif", "png");
$extension = end(explode(".", $_FILES["file"]["name"]));
if ((($_FILES["file"]["type"] == "image/gif")
|| ($_FILES["file"]["type"] == "image/jpeg")
|| ($_FILES["file"]["type"] == "image/png")
|| ($_FILES["file"]["type"] == "image/pjpeg"))
&& ($_FILES["file"]["size"] < 20000)
&& in_array($extension, $allowedExts))
{
    if ($_FILES["file"]["error"] > 0)
    {
        echo "Error: " . $_FILES["file"]["error"] . "<br>";
    }
    else
    {
        echo "Upload: " . $_FILES["file"]["name"] . "<br>";
        echo "Type: " . $_FILES["file"]["type"] . "<br>";
        echo "Size: " . ($_FILES["file"]["size"] / 1024) . " kB<br>";
        echo "Stored in: " . $_FILES["file"]["tmp_name"];
    }
}
else
{
    echo "Invalid file";
}
?>

```

Lưu ý: Đối với IE để nhận ra loại tập tin jpg phải để là pjpeg, cho FireFox nó phải là jpeg.

Saving the Uploaded File

Các ví dụ trên tạo ra một bản sao tạm thời của các tập tin tải lên trong thư mục temp PHP trên máy chủ.

Các file tạm thời sao chép mất đi ngay cả khi kịch bản kết thúc. Để lưu trữ các tập tin được tải lên chúng ta cần phải sao chép nó vào một vị trí khác nhau:

```

<?php
$allowedExts = array("jpg", "jpeg", "gif", "png");
$extension = end(explode(".", $_FILES["file"]["name"]));
if ((($_FILES["file"]["type"] == "image/gif")
|| ($_FILES["file"]["type"] == "image/jpeg")
|| ($_FILES["file"]["type"] == "image/png")
|| ($_FILES["file"]["type"] == "image/pjpeg"))
&& ($_FILES["file"]["size"] < 20000)
&& in_array($extension, $allowedExts))
{
    if ($_FILES["file"]["error"] > 0)
    {
        echo "Return Code: " . $_FILES["file"]["error"] . "<br>";
    }
    else
    {
        echo "Upload: " . $_FILES["file"]["name"] . "<br>";
        echo "Type: " . $_FILES["file"]["type"] . "<br>";
        echo "Size: " . ($_FILES["file"]["size"] / 1024) . " kB<br>";
        echo "Temp file: " . $_FILES["file"]["tmp_name"] . "<br>";

        if (file_exists("upload/" . $_FILES["file"]["name"]))
        {
            echo $_FILES["file"]["name"] . " already exists. ";
        }
        else
        {
            move_uploaded_file($_FILES["file"]["tmp_name"],
            "upload/" . $_FILES["file"]["name"]);
            echo "Stored in: " . "upload/" . $_FILES["file"]["name"];
        }
    }
}
else
{
    echo "Invalid file";
}
?>

```

Các kịch bản trên kiểm tra nếu tập tin đã tồn tại, nếu nó không có, nó sẽ sao chép các tập tin vào một thư mục gọi là "upload".

Chapter 20 : PHP Cookies

Một cookie thường được sử dụng để xác định một người sử dụng.

What is a Cookie?

Một cookie thường được sử dụng để xác định một người sử dụng. Một cookie là một file nhỏ mà server nhúng vào trên máy tính của người dùng. Mỗi lần cùng một máy tính yêu cầu một trang với một trình duyệt, nó sẽ gửi cookie. Với PHP, bạn có thể tạo ra và lấy giá trị cookie.

How to Create a Cookie?

Hàm `setcookie()` được sử dụng để thiết lập một cookie.

Lưu ý: Hàm `setcookie()` phải có TRƯỚC thẻ `<html>`.

Cú pháp

```
setcookie(name, value, expire, path, domain);
```

Ví dụ 1

Trong ví dụ dưới đây, chúng tôi sẽ tạo ra một cookie có tên là "user" và gán giá trị "Alex Porter" với nó. Chúng tôi cũng xác định rằng cookie hết hạn sau một giờ:

```
<?php
setcookie("user", "Alex Porter", time()+3600);
?>

<html>
.....
```

Lưu ý: Giá trị của cookie được tự động urlencoded khi gửi các tập tin cookie, và tự động giải mã khi nhận (để ngăn chặn URLencoding, sử dụng `setrawcookie()` thay thế).

Ví dụ 2

Bạn cũng có thể thiết lập thời gian hết hạn của cookie theo một cách khác. Nó có thể được dễ dàng hơn bằng cách sử dụng giây.

```
<?php
$expire=time()+60*60*24*30;
setcookie("user", "Alex Porter", $expire);
?>

<html>
.....
```

Trong ví dụ trên được thiết lập thời gian hết hạn một tháng (60 giây * 60 phút * 24 giờ * 30 ngày).

How to Retrieve a Cookie Value?

PHP biến \$ _COOKIE được sử dụng để lấy một giá trị cookie.

Trong ví dụ dưới đây, chúng tôi lấy giá trị của cookie có tên là "người sử dụng" và hiển thị nó trên một trang:

```
<?php
// Tạo 1 Cookie
echo $_COOKIE["user"];

// Cách để View Cookie
print_r($_COOKIE);
?>
```

Trong ví dụ sau, chúng tôi sử dụng isset() chức năng để tìm ra nếu một cookie đã được thiết lập:

```
<html>
<body>

<?php
if (isset($_COOKIE["user"]))
    echo "Welcome " . $_COOKIE["user"] . "!"<br>;
else
    echo "Welcome guest!"<br>;
?>
</body>
</html>
```

How to Delete a Cookie?

Khi xóa một cookie, bạn nên đảm bảo rằng ngày hết hạn là trong quá khứ.

Ví dụ xóa Cookie:

```
<?php
// set the expiration date to one hour ago
setcookie("user", "", time()-3600);
?>
```

What if a Browser Does NOT Support Cookies?

Nếu các chương trình khuyến mại áp dụng với các trình duyệt không hỗ trợ cookies, bạn sẽ phải sử dụng các phương pháp khác để vượt qua thông tin từ trang này sang trang khác trong ứng dụng của bạn. Một phương pháp là để truyền dữ liệu thông qua các hình thức (hình thức và đầu vào người sử dụng được mô tả trước đó trong hướng dẫn này).

Mẫu dưới đây qua các đầu vào người sử dụng "welcome.php" khi người dùng nhấp vào nút "Submit":

```
<html>
<body>

<form action="welcome.php" method="post">
Name: <input type="text" name="name">
Age: <input type="text" name="age">
<input type="submit">
</form>

</body>
</html>
```

Các giá trị trong tập tin "welcome.php" như thế này:

```
<html>
<body>

Welcome <?php echo $_POST["name"]; ?>.<br>
You are <?php echo $_POST["age"]; ?> years old.

</body>
</html>
```

Chapter 21 : PHP Sessions

Một biến PHP sessions được sử dụng để lưu trữ thông tin về, hoặc thay đổi cài đặt cho một phiên người dùng. Biến Session giữ thông tin về người sử dụng duy nhất, và có sẵn cho tất cả các trang trong một ứng dụng.

PHP Session Variables

Khi bạn đang làm việc với một ứng dụng, bạn mở nó, làm một số thay đổi và sau đó bạn đóng nó. Đây là giống như một phiên (sessions). Các máy tính biết bạn là ai. Nó biết khi bạn bắt đầu ứng dụng và khi bạn kết thúc. Tuy nhiên, trên internet có một vấn đề: các máy chủ web không biết bạn là ai và những gì bạn làm vì địa chỉ HTTP không duy trì trạng thái.

Một PHP session giải quyết vấn đề này bằng cách cho phép bạn lưu trữ thông tin người

dùng trên máy chủ để sử dụng sau này (ví dụ như tên người dùng, các mặt hàng mua sắm, vv). Tuy nhiên, thông tin phiên là tạm thời và sẽ bị xóa sau khi người dùng rời khỏi trang web. Nếu bạn cần lưu trữ vĩnh viễn, bạn có thể muốn để lưu trữ các dữ liệu trong một cơ sở dữ liệu.

Phiên làm việc bằng cách tạo ra một id duy nhất (UID) cho mỗi khách truy cập và các biến của hàng dựa trên UID này. UID là hoặc được lưu trữ trong một cookie hoặc là tuyên truyền trong URL.

Starting a PHP Session

Trước khi bạn có thể lưu trữ thông tin người sử dụng trong phiên (sessions) PHP của bạn, trước tiên bạn phải bắt đầu sessions.

Lưu ý: Các chức năng `session_start()` phải xuất hiện TRƯỚC thẻ `<html>`:

```
<?php session_start(); ?>

<html>
<body>

</body>
</html>
```

Đoạn mã trên sẽ đăng ký phiên làm việc của người dùng với máy chủ, cho phép bạn để bắt đầu thông tin người dùng tiết kiệm, và gán một UID cho session của người dùng đó.

Storing a Session Variable

Cách chính xác để lưu trữ và lấy biến phiên giao dịch là sử dụng biến `$_SESSION` PHP:

```

<?php
session_start();
// store session data
$_SESSION['views']=1;
?>

<html>
<body>

<?php
//retrieve session data
echo "Pageviews=". $_SESSION['views'];
?>

</body>
</html>

```

Kết quả cho ra là

```
Pageviews=1
```

Trong ví dụ dưới đây, chúng tôi tạo ra một truy cập trang page-views đơn giản. Các `isset()` chức năng kiểm tra nếu biến "views" đã được thiết lập. Nếu "views" đã được thiết lập, chúng ta có thể tăng truy cập của chúng tôi. Nếu "views" không tồn tại, chúng tôi tạo ra một biến "views", và đặt nó vào 1:

```

<?php
session_start();

if(isset($_SESSION['views']))
    $_SESSION['views']=$_SESSION['views']+1;
else
    $_SESSION['views']=1;
echo "Views=". $_SESSION['views'];
?>

```

Destroying a Session

Nếu bạn muốn xóa một số dữ liệu phiên (sessions), bạn có thể sử dụng `unset()` hoặc `session_destroy()` .

Unset() chức năng được sử dụng để giải phóng biến phiên giao dịch quy định:

```
<?php
session_start();
if(isset($_SESSION['views']))
    unset($_SESSION['views']);
?>
```

Bạn cũng có thể phá hủy hoàn toàn các session bằng cách gọi session_destroy() chức năng:

```
<?php
session_destroy();
?>
```

Lưu ý: session_destroy () sẽ thiết lập lại phiên làm việc của bạn và bạn sẽ mất tất cả các dữ liệu sessions của bạn được lưu trữ.

Chapter 22 : Sending Emails

PHP cho phép bạn gửi e-mail trực tiếp từ một đoạn script.

The PHP mail() Function

PHP mail() chức năng được sử dụng để gửi email từ bên trong một kịch bản (Một Script ý).

Cú pháp

```
mail(to,subject,message,headers,parameters)
```

Parameter	Description
to	Required. Specifies the receiver / receivers of the email
subject	Required. Specifies the subject of the email. Note: This parameter cannot contain any newline characters
message	Required. Defines the message to be sent. Each line should be separated with a LF (\n). Lines should not exceed 70 characters
headers	Optional. Specifies additional headers, like From, Cc, and Bcc. The additional headers should be separated with a CRLF (\r\n)
parameters	Optional. Specifies an additional parameter to the sendmail program

Lưu ý: Đối với các chức năng mail sẽ có sẵn, PHP đòi hỏi một hệ thống email được cài đặt và hoạt động. Chương trình được sử dụng được xác định bởi các thiết lập cấu hình trong file php.ini.

PHP Simple E-Mail

Cách đơn giản nhất để gửi một email với PHP là gửi một email văn bản.

Trong ví dụ dưới đây, chúng tôi lần đầu tiên khai báo các biến (\$to, \$subject, \$message, \$from, \$headers), sau đó chúng tôi sử dụng các biến trong hàm mail() để gửi e-mail:

```
<?php
$to = "someone@example.com";
$subject = "Test mail";
$message = "Hello! This is a simple email message.";
$from = "someoneelse@example.com";
$headers = "From:" . $from;
mail($to,$subject,$message,$headers);
echo "Mail Sent.";
?>
```

PHP Mail Form

With PHP, you can create a feedback-form on your website. The example below sends a text message to a specified e-mail address:

```

<html>
<body>

<?php
if (isset($_REQUEST['email']))
//if "email" is filled out, send email
{
    //send email
    $email = $_REQUEST['email'] ;
    $subject = $_REQUEST['subject'] ;
    $message = $_REQUEST['message'] ;
    mail("someone@example.com", $subject,
    $message, "From:" . $email);
    echo "Thank you for using our mail form";
}
else
//if "email" is not filled out, display the form
{
    echo "<form method='post' action='mailform.php'>
    Email: <input name='email' type='text'><br>
    Subject: <input name='subject' type='text'><br>
    Message:<br>
    <textarea name='message' rows='15' cols='40'>
    </textarea><br>
    <input type='submit'>
    </form>";
}
?>

</body>
</html>

```

đây là ví dụ trên làm việc như thế nào:

Trước tiên, hãy kiểm tra nếu trường email đầu vào sẽ được điền

Nếu nó không được thiết lập (như khi trang là lần đầu tiên truy cập), sản lượng dưới dạng HTML

Nếu nó được thiết lập (sau khi biểu mẫu được điền đầy đủ), gửi email từ hình thức

Khi submit được nhấn sau khi biểu mẫu được điền đầy đủ, trang tải lại, thấy rằng các đầu vào email được thiết lập, và gửi email

Lưu ý: Đây là cách đơn giản để gửi e-mail, nhưng nó không phải là an toàn. Trong

chương kế tiếp của hướng dẫn này, bạn có thể đọc thêm chi tiết về lỗ hổng trong kịch bản e-mail, và làm thế nào để xác nhận đầu vào người sử dụng để làm cho nó an toàn hơn.

Chapter 23 : Secure E-mails

Có một điểm yếu trong kịch bản e-mail PHP trong chương trước.

PHP E-mail Injections

Đầu tiên, hãy nhìn vào các mã PHP từ chương trước:

```
<html>
<body>

<?php
if (isset($_REQUEST['email']))
//if "email" is filled out, send email
{
    //send email
    $email = $_REQUEST['email'] ;
    $subject = $_REQUEST['subject'] ;
    $message = $_REQUEST['message'] ;
    mail("someone@example.com", "Subject: $subject",
    $message, "From: $email" );
    echo "Thank you for using our mail form";
}
else
//if "email" is not filled out, display the form
{
    echo "<form method='post' action='mailform.php'>
    Email: <input name='email' type='text'><br>
    Subject: <input name='subject' type='text'><br>
    Message:<br>
    <textarea name='message' rows='15' cols='40'>
    </textarea><br>
    <input type='submit'>
    </form>";
}
?>

</body>
</html>
```

Vấn đề với code ở trên là người sử dụng trái phép có thể chèn dữ liệu vào các tiêu đề mail thông qua hình thức đầu vào.

Điều gì sẽ xảy ra nếu người dùng thêm các văn bản sau đây đến lĩnh vực nhập địa chỉ email trong các hình thức?

Như thế này ?

```
someone@example.com%0ACc:person2@example.com
%0ABcc:person3@example.com,person3@example.com,
anotherperson4@example.com,person5@example.com
%0ABTo:person6@example.com
```

Hàm mail () đặt các văn bản trên vào các tiêu đề thư như bình thường, và phần đầu trang có thêm một Cc:, Bcc:, và: To : field. Khi người dùng nhấp vào nút submit, e-mail sẽ được gửi đến tất cả các địa chỉ trên!

PHP Stopping E-mail Injections

Cách tốt nhất để ngăn chặn khai thác e-mail là để xác nhận đầu vào.

Đoạn code dưới đây là giống như trong chương trước, nhưng bây giờ chúng tôi đã thêm một validator đầu vào để kiểm tra các lĩnh vực thư điện tử trong các hình thức:

```

<html>
<body>
<?php
function spamcheck($field)
{
    //filter_var() sanitizes the e-mail
    //address using FILTER_SANITIZE_EMAIL
    $field=filter_var($field, FILTER_SANITIZE_EMAIL);

    //filter_var() validates the e-mail
    //address using FILTER_VALIDATE_EMAIL
    if(filter_var($field, FILTER_VALIDATE_EMAIL))
    {
        return TRUE;
    }
    else
    {
        return FALSE;
    }
}
if (isset($_REQUEST['email']))
    {//if "email" is filled out, proceed

        //check if the email address is invalid
        $mailcheck = spamcheck($_REQUEST['email']);
        if ($mailcheck==FALSE)
        {
            echo "Invalid input";
        }
        else
        {
            //send email
            $email = $_REQUEST['email'] ;
            $subject = $_REQUEST['subject'] ;
            $message = $_REQUEST['message'] ;
            mail("someone@example.com", "Subject: $subject",
            $message, "From: $email" );
            echo "Thank you for using our mail form";
        }
    }
else
    {
        //if "email" is not filled out, display the form
        echo "<form method='post' action='mailform.php'>
        Email: <input name='email' type='text'><br>
        Subject: <input name='subject' type='text'><br>
        Message:<br>
        <textarea name='message' rows='15' cols='40'>
    }

```

```
</textarea><br>
<input type='submit'>
</form>";
}
?>

</body>
</html>
```

Trong đoạn mã trên, chúng tôi sử dụng PHP bộ lọc để xác nhận đầu vào:

Các bộ lọc FILTER_SANITIZE_EMAIL loại bỏ tất cả các ký tự bất hợp pháp e-mail từ một chuỗi

Các bộ lọc FILTER_VALIDATE_EMAIL xác nhận giá trị như là một địa chỉ e-mail

Bạn có thể đọc thêm về các bộ lọc trong chương Lọc PHP của chúng tôi..

http://w3schools.com/php/php_filter.asp

Chapter 24 : Error Handling

Việc xử lý lỗi mặc định trong PHP rất đơn giản. Một thông báo lỗi với tên tập tin, số dòng và một tin nhắn mô tả các lỗi được gửi đến trình duyệt.

PHP Error Handling

Khi tạo ra các kịch bản và các ứng dụng web, lỗi xử lý là một phần quan trọng. Nếu mã của bạn thiếu mã kiểm tra lỗi, chương trình của bạn có thể trông rất không chuyên nghiệp và bạn có thể có rủi ro an ninh.

Hướng dẫn này có chứa một số lỗi phổ biến nhất kiểm tra các phương pháp trong PHP.

Chúng tôi sẽ hiển thị các phương pháp xử lý các lỗi khác nhau:

- Đơn giản "die()" báo cáo
- Tùy chỉnh lỗi và gây nên lỗi
- báo cáo lỗi

Basic Error Handling: Using the die() function

Ví dụ đầu tiên cho thấy một kịch bản đơn giản mở một tập tin văn bản:

```
<?php
$file=fopen("welcome.txt","r");
?>
```

Nếu tập tin không tồn tại, bạn có thể nhận được một lỗi như thế này:

Warning: fopen(welcome.txt) [function.fopen]: failed to open stream:
No such file or directory in **C:\www\test.php** on line **2**

Để tránh mà người dùng nhận được một thông báo lỗi như trên, chúng tôi kiểm tra nếu tập tin tồn tại trước khi chúng tôi cố gắng để truy cập vào nó:

```
<?php
if(!file_exists("welcome.txt"))
{
    die("File not found");
}
else
{
    $file=fopen("welcome.txt","r");
}
?>
```

Bây giờ nếu các tập tin không tồn tại, bạn nhận được một lỗi như thế này:

File not found

Đoạn mã trên là hiệu quả hơn so với mã trước đó, bởi vì nó sử dụng một cơ chế xử lý lỗi đơn giản để ngăn chặn kịch bản sau khi lỗi.

Tuy nhiên, chỉ cần dừng lại kịch bản không phải là luôn luôn đúng cách để đi. Chúng ta hãy xem xét thay thế PHP chức năng cho các lỗi xử lý.

Creating a Custom Error Handler

Tạo một lỗi xử lý tùy chỉnh là khá đơn giản. Chúng tôi chỉ đơn giản là tạo ra một chức năng đặc biệt có thể được gọi khi lỗi xảy ra trong PHP.

Chức năng này phải có khả năng để xử lý tối thiểu là hai tham số (mức độ lỗi và thông báo lỗi), nhưng có thể chấp nhận đến năm tham số (tùy chọn tập tin, đường số, và bối cảnh các lỗi);

Cú pháp

```
error_function(error_level,error_message,  
error_file,error_line,error_context)
```

Parameter	Description
error_level	Required. Specifies the error report level for the user-defined error. Must be a value number. See table below for possible error report levels
error_message	Required. Specifies the error message for the user-defined error
error_file	Optional. Specifies the filename in which the error occurred
error_line	Optional. Specifies the line number in which the error occurred
error_context	Optional. Specifies an array containing every variable, and their values, in use when the error occurred

Error Report levels

Những mức độ báo cáo lỗi là các loại khác nhau của các lỗi người dùng định nghĩa xử lý lỗi có thể được sử dụng cho:

Value	Constant	Description
2	E_WARNING	Non-fatal run-time errors. Execution of the script is not halted
8	E_NOTICE	Run-time notices. The script found something that might be an error, but could also happen when running a script normally

256	E_USER_ERROR	Fatal user-generated error. This is like an E_ERROR set by the programmer using the PHP function trigger_error()
512	E_USER_WARNING	Non-fatal user-generated warning. This is like an E_WARNING set by the programmer using the PHP function trigger_error()
1024	E_USER_NOTICE	User-generated notice. This is like an E_NOTICE set by the programmer using the PHP function trigger_error()
4096	E_RECOVERABLE_ERROR	Catchable fatal error. This is like an E_ERROR but can be caught by a user defined handle (see also set_error_handler())
8191	E_ALL	All errors and warnings (E_STRICT became a part of E_ALL in PHP 5.4)

Bây giờ cho phép tạo ra một chức năng để xử lý các lỗi:

```
function customError($errno, $errstr)
{
    echo "<b>Error:</b> [$errno] $errstr<br>";
    echo "Ending Script";
    die();
}
```

Đoạn mã trên là một lỗi chức năng xử lý đơn giản. Khi nó được kích hoạt, nó được mức độ lỗi và một thông báo lỗi. Sau đó kết quả đầu ra mức độ lỗi và thông điệp và kết thúc kịch bản.

Bây giờ chúng ta đã tạo ra một chức năng xử lý lỗi, chúng tôi cần phải quyết định khi nó cần phải được kích hoạt.

Set Error Handler

Xử lý lỗi mặc định cho PHP là được xây dựng trong xử lý lỗi. Chúng tôi sẽ thực hiện các chức năng trên xử lý lỗi mặc định trong suốt thời gian của kịch bản.

Nó có thể thay đổi xử lý lỗi để áp dụng chỉ cho một số lỗi, như vậy kịch bản có thể xử lý các lỗi khác nhau theo những cách khác nhau. Tuy nhiên, trong ví dụ này, chúng ta sẽ sử dụng xử lý lỗi tùy chỉnh của chúng tôi cho tất cả các lỗi:

```
set_error_handler("customError");
```

Từ khi chúng tôi muốn có chức năng tùy chỉnh của chúng tôi để xử lý tất cả các lỗi, `set_error_handler()` chỉ cần một tham số, một tham số thứ hai có thể được thêm vào để xác định một mức độ lỗi.

Ví dụ

Kiểm tra xử lý lỗi bằng cách cố gắng để biến đầu ra không tồn tại:

```
<?php
//error handler function
function customError($errno, $errstr)
{
    echo "<b>Error:</b> [$errno] $errstr";
}

//set error handler
set_error_handler("customError");

//trigger error
echo($test);
?>
```

Đầu ra của các mã trên nên được một cái gì đó như thế này:

```
Error: [8] Undefined variable: test
```

Trigger an Error

Trong một kịch bản mà người sử dụng có thể nhập dữ liệu nó rất hữu ích để kích hoạt các lỗi khi một đầu vào bất hợp pháp xảy ra. Trong PHP, điều này được thực hiện bởi các chức năng `trigger_error()`.

Ví dụ

Trong ví dụ này, một lỗi xảy ra nếu biến "test" lớn hơn "1":

```
<?php
$test=2;
if ($test>1)
{
trigger_error("Value must be 1 or below");
}
?>
```

Đầu ra của các mã trên nên được một cái gì đó như thế này:

Notice: Value must be 1 or below
in **C:\webfolder\test.php** on line **6**

Một lỗi có thể được kích hoạt bất cứ nơi nào bạn muốn trong một kịch bản, và bằng cách thêm vào một tham số thứ hai, bạn có thể xác định mức độ lỗi được kích hoạt.

Loại lỗi có thể xảy ra:

E_USER_ERROR - Fatal thời gian chạy lỗi do người dùng tạo ra. Lỗi mà không thể được phục hồi từ. Thực thi kịch bản phải dừng lại

E_USER_WARNING - Không gây hư cảnh báo người dùng tạo ra thời gian chạy. Thực thi kịch bản không dừng lại

E_USER_NOTICE - Default. Người sử dụng tạo thông báo thời gian chạy. Các kịch bản tìm thấy một cái gì đó có thể là một lỗi, nhưng cũng có thể xảy ra khi chạy một kịch bản bình thường

Ví dụ

Trong ví dụ này E_USER_WARNING xảy ra nếu biến "test" lớn hơn "1". Nếu E_USER_WARNING xảy ra, chúng ta sẽ sử dụng tùy chỉnh xử lý lỗi của chúng tôi và kết thúc kịch bản:

```
<?php
//error handler function
function customError($errno, $errstr)
{
    echo "<b>Error:</b> [$errno] $errstr<br>";
    echo "Ending Script";
    die();
}

//set error handler
set_error_handler("customError",E_USER_WARNING);

//trigger error
$test=2;
if ($test>1)
{
    trigger_error("Value must be 1 or below",E_USER_WARNING);
}
?>
```

Đầu ra của các mã trên nên được một cái gì đó như thế này:

```
Error: [512] Value must be 1 or below
Ending Script
```

Bây giờ chúng ta đã học được cách tạo ra các lỗi của chúng ta và làm thế nào để kích hoạt chúng, cho phép có một cái nhìn lỗi đăng nhập.

Error Logging

Theo mặc định, PHP sẽ gửi một bản ghi lỗi hệ thống đăng nhập của máy chủ hoặc một tập tin, tùy thuộc vào cách cấu hình `error_log` được thiết lập trong file `php.ini`. Bằng cách sử dụng các `error_log()` chức năng bạn có thể gửi các bản ghi lỗi vào một tập tin được chỉ định hoặc một điểm đến từ xa.

Gửi thông báo lỗi cho chính mình bằng e-mail có thể là một cách tốt để nhận được thông báo lỗi cụ thể.

Send an Error Message by E-Mail

Trong ví dụ dưới đây, chúng tôi sẽ gửi một e-mail với một thông báo lỗi và kết thúc kịch bản, nếu một lỗi cụ thể xảy ra:

```

<?php
//error handler function
function customError($errno, $errstr)
{
    echo "<b>Error:</b> [$errno] $errstr<br>";
    echo "Webmaster has been notified";
    error_log("Error: [$errno] $errstr",1,
    "someone@example.com","From: webmaster@example.com");
}

//set error handler
set_error_handler("customError",E_USER_WARNING);

//trigger error
$test=2;
if ($test>1)
{
    trigger_error("Value must be 1 or below",E_USER_WARNING);
}
?>

```

Đầu ra của các mã trên nên được một cái gì đó như thế này:

Error: [512] Value must be 1 or below
Webmaster has been notified

Và các thư nhận được từ các mã trên trông như thế này:

Error: [512] Value must be 1 or below

Điều này không nên được sử dụng với tất cả các lỗi. Các lỗi thường xuyên nên được đăng nhập trên máy chủ bằng cách sử dụng hệ thống mặc định khai thác gổ PHP.

Chapter 25 : Exception Handling

Trường hợp ngoại lệ (Exception) được sử dụng để thay đổi dòng chảy bình thường của một kịch bản nếu một lỗi được xác định xảy ra.

What is an Exception

Với PHP 5, đến một đối tượng mới theo định hướng cách để đối phó với các lỗi.

Xử lý ngoại lệ được sử dụng để thay đổi dòng chảy bình thường của việc thực thi mã nếu một lỗi được chỉ định (đặc biệt) điều kiện xảy ra. Tình trạng này được gọi là một ngoại lệ.

Đây là những gì thường xảy ra khi một ngoại lệ được kích hoạt:

- Mã hiện tại của nhà nước được lưu
- Việc thực thi mã sẽ chuyển sang một chức năng được xác định trước (tùy chỉnh) xử lý ngoại lệ
- Tùy theo tình hình, xử lý sau đó có thể tiếp tục thực hiện từ trạng thái mã lưu, chấm dứt thực hiện kịch bản hoặc tiếp tục kịch bản từ một vị trí khác nhau trong các mã

Chúng tôi sẽ hiển thị các phương pháp xử lý các lỗi khác nhau:

- Sử dụng cơ bản của ngoại lệ
- Tạo một xử lý ngoại lệ tùy chỉnh nhiều trường hợp ngoại lệ
- Lại ném một ngoại lệ
- Thiết lập một xử lý ngoại lệ đầu

Lưu ý: Trường hợp ngoại lệ chỉ nên được sử dụng với điều kiện lỗi, và không nên được sử dụng để chuyển đến một nơi khác trong các mã tại một thời điểm xác định.

Basic Use of Exceptions

Khi một ngoại lệ được ném ra, các mã sau nó sẽ không được thực hiện, và PHP sẽ cố gắng để tìm thấy những "catch" phù hợp với khối.

Nếu một ngoại lệ không bị bắt, một lỗi nghiêm trọng sẽ được phát hành với một thông báo "uncaught ngoại lệ".

Cho phép thử để ném một ngoại lệ mà không đánh bắt nó:

```
<?php
//create function with an exception
function checkNum($number)
{
    if($number>1)
    {
        throw new Exception("Value must be 1 or below");
    }
    return true;
}

//trigger exception
checkNum(2);
?>
```

Đoạn mã trên sẽ nhận được một lỗi như thế này:

```
Fatal error: Uncaught exception 'Exception'
with message 'Value must be 1 or below' in C:\webfolder\test.php:6
Stack trace: #0 C:\webfolder\test.php(12):
checkNum(28) #1 {main} thrown in C:\webfolder\test.php on line 6
```

Try, throw and catch

Để tránh lỗi này từ ví dụ trên, chúng ta cần để tạo ra các mã thích hợp để xử lý một trường hợp ngoại lệ.

Thích hợp ngoại lệ mã nên bao gồm:

1. Cố gắng - Một chức năng bằng cách sử dụng một ngoại lệ phải ở trong một khối "thử". Nếu trường hợp ngoại lệ không kích hoạt, mã sẽ tiếp tục như bình thường. Tuy nhiên, nếu trường hợp ngoại lệ kích hoạt, một ngoại lệ là "ném"
2. Throw - Đây là cách bạn kích hoạt một ngoại lệ. Mỗi "ném" phải có ít nhất một "catch"
3. Catch - Một khối "bắt" lấy một ngoại lệ và tạo ra một đối tượng có chứa các thông

tin ngoại lệ

```
<?php
//create function with an exception
function checkNum($number)
{
    if($number>1)
    {
        throw new Exception("Value must be 1 or below");
    }
    return true;
}

//trigger exception in a "try" block
try
{
    checkNum(2);
    //If the exception is thrown, this text will not be shown
    echo 'If you see this, the number is 1 or below';
}

//catch exception
catch(Exception $e)
{
    echo 'Message: ' . $e->getMessage();
}
?>
```

Đoạn mã trên sẽ nhận được một lỗi như thế này:

Message: Value must be 1 or below

Example explained:

Đoạn mã trên ném một ngoại lệ và bắt nó:

1. Các checkNum() chức năng được tạo ra. Nó kiểm tra nếu một số là số lớn hơn 1. Nếu

có, một ngoại lệ được ném

2. Các `checkNum()` chức năng được gọi trong một khối "thử"

3. Ngoại trừ trong phạm vi chức năng `checkNum()` được ném

4. Block "bắt" retrieves các trường hợp ngoại lệ và tạo ra một đối tượng (`$e`) có chứa các thông tin ngoại lệ

5. Các thông báo lỗi từ trường hợp ngoại lệ được lặp lại bằng cách gọi `$e->getMessage()` từ đối tượng ngoại lệ

Tuy nhiên, một trong những cách để có được xung quanh "ném mỗi phải có một nắm bắt" nguyên tắc là để thiết lập một mức độ ngoại lệ xử lý hàng đầu để xử lý các lỗi phiếu thông qua.

Creating a Custom Exception Class

Tạo một xử lý ngoại lệ tùy chỉnh là khá đơn giản. Chúng tôi chỉ đơn giản là tạo ra một lớp học đặc biệt với chức năng có thể được gọi khi một ngoại lệ xảy ra trong PHP. Các lớp học phải là một phần mở rộng của lớp ngoại lệ.

Các lớp ngoại lệ tùy chỉnh được thừa hưởng tài sản từ lớp ngoại lệ của PHP và bạn có thể thêm các chức năng tùy chỉnh cho nó.

Cho phép tạo ra một lớp ngoại lệ:

```
<?php
class customException extends
Exception
{
    public function errorMessage()
    {
        //error message
        $errorMsg = 'Error on line '.$this-
>getLine().' in '.$this->getFile()
        .': <b>'.$this->getMessage().</b> is
not a valid E-Mail address';
        return $errorMsg;
    }
}

$email = "someone@example...com";

try
{
    //check if
    if(filter_var($email,
FILTER_VALIDATE_EMAIL) ===
FALSE)
```

```

    {
        //throw exception if email is not
        valid
        throw new
        customException($email);
    }
}

catch (customException $e)
{
    //display custom message
    echo $e->errorMessage();
}
?>

```

Các lớp học mới là một bản sao của lớp ngoại lệ cũ với một sự bổ sung chức năng ErrorMessage(). Vì nó là một bản sao của lớp cũ, và nó được thừa hưởng các thuộc tính và phương pháp từ lớp cũ, chúng ta có thể sử dụng các phương thức lớp ngoại lệ như getline() và getFile() và getMessage().

Example explained:

Đoạn mã trên ném một ngoại lệ và bắt nó với một lớp ngoại lệ tùy chỉnh:

- 1.Lớp customException() được tạo ra như là một phần mở rộng của lớp ngoại lệ cũ. Bằng cách này, nó được thừa hưởng tất cả các phương pháp và các thuộc tính từ lớp ngoại lệ cũ
 - 2.Chức năng ErrorMessage() được tạo ra. Hàm này trả về một thông báo lỗi nếu một địa chỉ e-mail không hợp lệ
 - 3.Biến \$email được thiết lập để một chuỗi đó không phải là một địa chỉ e-mail hợp lệ
 - 4."Thử" khối được thực hiện và ngoại lệ được ném từ địa chỉ e-mail không hợp lệ
 - 5.Khối "catch" bắt những trường hợp ngoại lệ và hiển thị các thông báo lỗi
-

Multiple Exceptions

Nó có thể cho một kịch bản để sử dụng nhiều trường hợp ngoại lệ để kiểm tra nhiều điều kiện.

Có thể sử dụng một số nếu .. khác khối, một chuyển đổi, hoặc nhiều trường hợp ngoại lệ làm tổ. Những trường hợp ngoại lệ này có thể sử dụng các lớp học ngoại lệ khác nhau và trả lại thông báo lỗi khác nhau:

```

<?php
class customException extends Exception
{
public function errorMessage()
{
//error message
$errorMsg = 'Error on line '.$this->getLine().' in '.$this->getFile()
.': <b>'.$this->getMessage().'</b> is not a valid E-Mail address';
return $errorMsg;
}
}

$email = "someone@example.com";

try
{
//check if
if(filter_var($email, FILTER_VALIDATE_EMAIL) === FALSE)
{
//throw exception if email is not valid
throw new customException($email);
}
//check for "example" in mail address
if(strpos($email, "example") !== FALSE)
{
throw new Exception("$email is an example e-mail");
}
}

catch (customException $e)
{
echo $e->errorMessage();
}

catch(Exception $e)
{
echo $e->getMessage();
}
?>

```

Example explained:

Đoạn mã trên kiểm tra hai điều kiện và ném một ngoại lệ nếu có các điều kiện không được đáp ứng:

1. Lớp `customException ()` được tạo ra như là một phần mở rộng của lớp ngoại lệ cũ. Bằng cách này, nó được thừa hưởng tất cả các phương pháp và các thuộc tính từ lớp ngoại lệ cũ
2. Chức năng `ErrorMessage ()` được tạo ra. Hàm này trả về một thông báo lỗi nếu một địa chỉ e-mail không hợp lệ
3. Biến `$ email` được thiết lập để một chuỗi đó là một địa chỉ e-mail hợp lệ, nhưng có chứa chuỗi "ví dụ"
4. "Thử" khối được thực hiện và một ngoại lệ được ném vào điều kiện đầu tiên
5. Điều kiện thứ hai gây ra một ngoại lệ kể từ khi e-mail có chứa chuỗi "ví dụ"
6. Khối "catch" bắt những trường hợp ngoại lệ và hiển thị các thông báo lỗi chính xác

Nếu ném ngoại lệ của `customException` lớp và không có bắt `customException`, chỉ có ngoại lệ cơ sở đánh bắt, một ngoại lệ sẽ được xử lý ở đó.

Re-throwing Exceptions

Đôi khi, khi một ngoại lệ được ném ra, bạn có thể muốn để xử lý nó khác với cách tiêu chuẩn. Nó có thể ném một ngoại lệ lần thứ hai trong vòng một khối "catch".

Một kịch bản nên che giấu lỗi hệ thống từ người dùng. Lỗi hệ thống có thể quan trọng đối với các coder, nhưng không có người dùng quan tâm. Để làm cho mọi việc dễ dàng hơn cho người sử dụng, bạn có thể ném lại ngoại lệ với một thông điệp thân thiện với người sử dụng:

```

<?php
class customException extends Exception
{
    public function errorMessage()
    {
        //error message
        $errorMsg = $this->getMessage().' is not a valid E-Mail address.';
        return $errorMsg;
    }
}

$email = "someone@example.com";

try
{
    try
    {
        //check for "example" in mail address
        if(strpos($email, "example") !== FALSE)
        {
            //throw exception if email is not valid
            throw new Exception($email);
        }
    }
    catch(Exception $e)
    {
        //re-throw exception
        throw new customException($email);
    }
}

catch (customException $e)
{
    //display custom message
    echo $e->errorMessage();
}
?>

```

Example explained:

Đoạn mã trên bài kiểm tra nếu địa chỉ email có chứa chuỗi "ví dụ" trong đó, nếu có, các trường hợp ngoại lệ được lại ném:

1. Lớp `customException()` được tạo ra như là một phần mở rộng của lớp ngoại lệ cũ. Bằng cách này, nó được thừa hưởng tất cả các phương pháp và các thuộc tính từ lớp ngoại lệ cũ

2. Chức năng `ErrorMessage()` được tạo ra. Hàm này trả về một thông báo lỗi nếu một địa chỉ e-mail không hợp lệ

3. Biến `$email` được thiết lập để một chuỗi đó là một địa chỉ e-mail hợp lệ, nhưng có chứa chuỗi "ví dụ"

4. "Thử" khối chứa một "thử" khối để làm cho nó có thể ném ngoại lệ

5. Ngoại lệ này được kích hoạt kể từ khi e-mail có chứa chuỗi "ví dụ"

6. Khối "catch" bắt ngoại lệ và lại ném một "customException"

7. Bị bắt "customException" và hiển thị một thông báo lỗi

Nếu trường hợp ngoại lệ không được đánh bắt trong khối hiện tại của nó "thử", nó sẽ tìm kiếm một khối catch "cấp cao hơn".

Set a Top Level Exception Handler

`set_exception_handler()` chức năng thiết lập một chức năng người dùng định nghĩa để xử lý tất cả các trường hợp ngoại lệ `uncaught`.

```
<?php
function myException($exception)
{
    echo "<b>Exception:</b> " , $exception->getMessage();
}

set_exception_handler('myException');

throw new Exception('Uncaught Exception occurred');
?>
```

Đầu ra của các mã trên nên được một cái gì đó như thế này:

Exception: Uncaught Exception occurred

Trong đoạn mã trên không có block "catch". Thay vào đó, xử lý ngoại lệ cấp cao nhất kích hoạt. Chức năng này nên được sử dụng để bắt ngoại lệ `uncaught`.

Quy tắc cho trường hợp ngoại lệ

Mã có thể được bao bọc trong một khối thử, để giúp bắt ngoại lệ tiềm năng

Mỗi khối try hoặc "ném" phải có ít nhất một khối catch tương ứng

Nhiều khối catch có thể được sử dụng để bắt các lớp học khác nhau của các trường hợp

ngoại lệ

Trường hợp ngoại lệ có thể được ném (hoặc tái ném) trong một khối catch trong một khối try

Một nguyên tắc đơn giản: Nếu bạn ném một cái gì đó, bạn phải nắm bắt nó.

Chapter 26 : PHP Filter

PHP bộ lọc được sử dụng để xác nhận và lọc dữ liệu đến từ các nguồn không an toàn, giống như đầu vào người sử dụng.

What is a PHP Filter?

Một bộ lọc PHP được sử dụng để xác nhận và lọc dữ liệu đến từ các nguồn không an toàn.

Để kiểm tra, xác nhận và lọc đầu vào người dùng hoặc dữ liệu tùy chỉnh là một phần quan trọng của bất kỳ ứng dụng web.

Phần mở rộng PHP bộ lọc được thiết kế để lọc dữ liệu dễ dàng hơn và nhanh hơn.

Why use a Filter?

Hầu như tất cả các ứng dụng web phụ thuộc vào đầu vào bên ngoài. Thông thường, điều này xuất phát từ một người dùng hoặc một ứng dụng khác (như một dịch vụ web). Bằng cách sử dụng bộ lọc, bạn có thể chắc chắn rằng ứng dụng của bạn được các loại đầu vào chính xác.

Bạn nên luôn luôn lọc tất cả dữ liệu bên ngoài!

Đầu vào lọc là một trong những vấn đề bảo mật ứng dụng quan trọng nhất.

Dữ liệu bên ngoài là gì?

- Dữ liệu đầu vào từ biểu mẫu
 - Cookies
 - Web dịch vụ dữ liệu
 - máy chủ các biến
 - Cơ sở dữ liệu kết quả truy vấn
-

Functions and Filters

Để lọc một biến, sử dụng một trong các chức năng lọc sau đây:

- filter_var() - Bộ lọc một biến duy nhất với một bộ lọc xác định
- filter_var_array() - Bộ lọc một số biến với các bộ lọc giống và khác nhau
- filter_input - một biến đầu vào và lọc nó
- filter_input_array - Nhận một số biến đầu vào và lọc chúng với các bộ lọc giống và khác nhau

Trong ví dụ dưới đây, chúng tôi xác nhận một số nguyên bằng cách sử dụng chức năng filter_var():

```
<?php
$int = 123;

if(!filter_var($int, FILTER_VALIDATE_INT))
{
    echo("Integer is not valid");
}
else
{
    echo("Integer is valid");
}
?>
```

Chapter 27 : PHP MySQL

Cơ sở dữ liệu MySQL miễn phí là rất thường được sử dụng với PHP.

Tạo một kết nối đến một cơ sở dữ liệu MySQL

Trước khi bạn có thể truy cập dữ liệu trong một cơ sở dữ liệu, bạn phải tạo ra một kết nối đến cơ sở dữ liệu.

Trong PHP, điều này được thực hiện với các chức năng mysql_connect ().

Cú pháp

```
mysql_connect (servername, username, password) ;
```

Thông số	Mô tả
servername	Tùy chọn. Chỉ định máy chủ để kết nối với. Giá trị mặc định là "localhost: 3306"
Tên truy nhập	Tùy chọn. Chỉ định tên người dùng đăng nhập với. Giá trị mặc định là tên của người sử dụng sở hữu quá trình máy chủ

mật khẩu	Tùy chọn. Chỉ định mật khẩu để đăng nhập. Mặc định là ""
----------	--

Lưu ý: Có thêm nhiều thông số có sẵn, nhưng những người được liệt kê ở trên là quan trọng nhất. Đầy đủ của chúng tôi [PHP MySQL tham khảo](#) để biết thêm chi tiết.

Ví dụ

Trong ví dụ sau chúng tôi lưu trữ các kết nối trong một biến (\$con) cho sử dụng sau này trong kịch bản. "Chết" một phần sẽ được thực hiện nếu kết nối không thành công:

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

// some code
?>
```

Kết thúc một kết nối

Kết nối sẽ được tự động đóng khi kịch bản kết thúc. Để đóng kết nối trước khi sử dụng `mysql_close()` chức năng:

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

// some code

mysql_close($con);
?>
```

Một cơ sở dữ liệu chứa một hoặc nhiều bảng.

Tạo một cơ sở dữ liệu

Câu lệnh `CREATE DATABASE` được sử dụng để tạo ra một cơ sở dữ liệu trong MySQL.

Cú pháp

```
CREATE DATABASE database_name
```

Để tìm hiểu thêm về SQL, vui lòng truy cập của chúng tôi [hướng dẫn SQL](#).

Để có được PHP để thực hiện các tuyên bố trên, chúng ta phải sử dụng hàm `mysql_query()`. Chức năng này được sử dụng để gửi một truy vấn hoặc lệnh để kết nối MySQL.

Ví dụ

Ví dụ sau đây tạo ra một cơ sở dữ liệu gọi là "my_db":

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

if (mysql_query("CREATE DATABASE my_db",$con))
{
    echo "Database created";
}
else
{
    echo "Error creating database: " . mysql_error();
}

mysql_close($con);
?>
```

Tạo một bảng

Câu lệnh `CREATE TABLE` được sử dụng để tạo ra một bảng trong MySQL.

Cú pháp

```
CREATE TABLE table_name
(
    column_name1 data_type,
    column_name2 data_type,
    column_name3 data_type,
    ....
)
```

Để tìm hiểu thêm về SQL, vui lòng truy cập của chúng tôi [hướng dẫn SQL](#) .

Chúng ta phải thêm các tuyên bố `CREATE TABLE` để `mysql_query()` chức năng để thực hiện lệnh.

Ví dụ

Ví dụ sau đây tạo ra một bảng có tên là "Người", với ba cột. Các tên cột sẽ được "FirstName", "LastName" và "Age":

```

<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

// Create database
if (mysql_query("CREATE DATABASE my_db", $con))
{
    echo "Database created";
}
else
{
    echo "Error creating database: " . mysql_error();
}

// Create table
mysql_select_db("my_db", $con);
$sql = "CREATE TABLE Persons
(
    FirstName varchar(15),
    LastName varchar(15),
    Age int
)";

// Execute query
mysql_query($sql, $con);

mysql_close($con);
?>

```

Quan trọng: Một cơ sở dữ liệu phải được lựa chọn trước khi một bảng có thể được tạo ra. Cơ sở dữ liệu được chọn với `mysql_select_db ()` chức năng.

Lưu ý: Khi bạn tạo ra một lĩnh vực cơ sở dữ liệu kiểu `varchar`, bạn phải xác định độ dài tối đa của lĩnh vực này, ví dụ như `varchar (15)`.

Các kiểu dữ liệu xác định kiểu dữ liệu cột có thể giữ. Đối với một tài liệu tham khảo đầy đủ của tất cả các loại dữ liệu có sẵn trong MySQL, chúng tôi hoàn thành [tài liệu tham khảo kiểu dữ liệu](#) .

Khóa chính và các lĩnh vực Tăng Auto

Mỗi bảng cần phải có một trường tiểu học trọng điểm.

Một khóa chính được sử dụng để xác định các hàng trong một bảng. Mỗi giá trị khóa chính phải là duy nhất trong bảng. Hơn nữa, các trường tiểu học trọng điểm không có thể là vô giá trị vì các công cụ cơ sở dữ liệu đòi hỏi phải có một giá trị để xác định vị trí các bản ghi.

Ví dụ sau đây thiết lập các lĩnh vực `PersonId` như các lĩnh vực khóa chính. Các lĩnh vực trọng điểm chính thường là một số ID, và thường được sử dụng với các thiết lập

AUTO_INCREMENT. AUTO_INCREMENT tự động làm tăng giá trị của trường 1 mỗi lần một kỷ lục mới được thêm vào. Để đảm bảo rằng các trường tiểu học trọng điểm không thể null, chúng ta phải thêm các thiết lập NOT NULL cho lĩnh vực này.

Ví dụ

```
$sql = "CREATE TABLE Persons
(
personID int NOT NULL AUTO_INCREMENT,
PRIMARY KEY(personID),
FirstName varchar(15),
LastName varchar(15),
Age int
) ";
mysql_query($sql,$con);
```

Chèn dữ liệu vào một bảng cơ sở dữ liệu

INSERT INTO tuyên bố được sử dụng để thêm các bản ghi mới vào một bảng cơ sở dữ liệu.

Cú pháp

Có thể viết INSERT INTO tuyên bố trong hai hình thức.

Các hình thức đầu tiên không xác định các cột dữ liệu sẽ được chèn vào, chỉ có giá trị của họ:

```
INSERT INTO table_name
VALUES (value1, value2, value3,...)
```

Hình thức thứ hai quy định cụ thể cả các tên cột và các giá trị sẽ được chèn vào:

```
INSERT INTO table_name (column1, column2, column3,...)
VALUES (value1, value2, value3,...)
```

Để tìm hiểu thêm về SQL, vui lòng truy cập của chúng tôi [hướng dẫn SQL](#) .

Để có được PHP để thực hiện báo cáo trên, chúng ta phải sử dụng hàm mysql_query (). Chức năng này được sử dụng để gửi một truy vấn hoặc lệnh để kết nối MySQL.

Ví dụ

Trong chương trước, chúng ta đã tạo ra một bảng có tên là "Người", với ba cột "FirstName", "LastName" và "Age". Chúng tôi sẽ sử dụng cùng một bảng trong ví dụ này. Ví dụ sau đây cho biết thêm hai kỷ lục mới vào bảng "Persons":

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
```

```

{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

mysql_query("INSERT INTO Persons (FirstName, LastName, Age)
VALUES ('Peter', 'Griffin', 35)");

mysql_query("INSERT INTO Persons (FirstName, LastName, Age)
VALUES ('Glenn', 'Quagmire', 33)");

mysql_close($con);
?>

```

Chèn dữ liệu từ form vào cơ sở dữ liệu

Bây giờ chúng ta sẽ tạo ra một hình thức HTML có thể được sử dụng để thêm các hồ sơ mới để bảng "Người".

Đây là hình thức HTML:

```

<html>
<body>

<form action="insert.php" method="post">
Firstname: <input type="text" name="firstname">
Lastname: <input type="text" name="lastname">
Age: <input type="text" name="age">
<input type="submit">
</form>

</body>
</html>

```

Khi người dùng nhấp chuột vào nút gửi ở dạng HTML trong ví dụ trên, các dữ liệu mẫu được gửi "insert.php".

Các tập tin "insert.php" kết nối với một cơ sở dữ liệu, và lấy các giá trị từ hình thức với các biến PHP \$_POST.

Sau đó, mysql_query () chức năng thực hiện các INSERT INTO tuyên bố, và một kỷ lục mới sẽ được thêm vào bảng "Người".

Đây là trang "insert.php":

```

<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

```

```

    }

mysql_select_db("my_db", $con);

$sql="INSERT INTO Persons (FirstName, LastName, Age)
VALUES
('$_POST[firstname]', '$_POST[lastname]', '$_POST[age]')";

if (!mysql_query($sql,$con))
{
    die('Error: ' . mysql_error());
}
echo "1 record added";

mysql_close($con);
?>

```

Chọn dữ liệu từ một bảng cơ sở dữ liệu

Câu lệnh SELECT được sử dụng để chọn dữ liệu từ một cơ sở dữ liệu.

Cú pháp

```

SELECT column_name(s)
FROM table_name

```

Để tìm hiểu thêm về SQL, vui lòng truy cập của chúng tôi [hướng dẫn SQL](#) .

Để có được PHP để thực hiện các tuyên bố trên, chúng ta phải sử dụng hàm mysql_query (). Chức năng này được sử dụng để gửi một truy vấn hoặc lệnh để kết nối MySQL.

Ví dụ

Ví dụ sau đây chọn tất cả các dữ liệu được lưu trữ trong bảng "Người" (ký tự * chọn tất cả các dữ liệu trong bảng):

```

<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

$result = mysql_query("SELECT * FROM Persons");

while($row = mysql_fetch_array($result))
{
    echo $row['FirstName'] . " " . $row['LastName'];
    echo "<br />";
}

```

```
mysql_close($con);  
?>
```

Ví dụ trên cửa hàng dữ liệu trả về bởi hàm `mysql_query()` trong biến `$` kết quả.

Tiếp theo, chúng tôi sử dụng `mysql_fetch_array()` chức năng để trả lại hàng đầu tiên từ recordset như là một mảng. Mỗi cuộc gọi đến `mysql_fetch_array()` trả lại hàng kế tiếp trong recordset. Các vòng lặp trong khi vòng qua tất cả các bản ghi trong recordset. Để in giá trị của mỗi hàng, chúng tôi sử dụng biến `$row` PHP (`$row['FirstName']` và `$row['LastName']`).

Đầu ra của các mã trên sẽ là:

```
Peter Griffin  
Glenn Quagmire
```

Hiển thị các kết quả trong một bảng HTML

Ví dụ sau đây chọn các dữ liệu tương tự như ví dụ trên, nhưng sẽ hiển thị dữ liệu trong một bảng HTML:

```
<?php  
$con = mysql_connect("localhost","peter","abc123");  
if (!$con)  
{  
    die('Could not connect: ' . mysql_error());  
}  
  
mysql_select_db("my_db", $con);  
  
$result = mysql_query("SELECT * FROM Persons");  
  
echo "<table border='1'>  
<tr>  
<th>Firstname</th>  
<th>Lastname</th>  
</tr>";  
  
while($row = mysql_fetch_array($result))  
{  
    echo "<tr>";  
    echo "<td>" . $row['FirstName'] . "</td>";  
    echo "<td>" . $row['LastName'] . "</td>";  
    echo "</tr>";  
}  
echo "</table>";  
  
mysql_close($con);  
?>
```

Đầu ra của các mã trên sẽ là:

FirstName	Lastname
Glenn	Bai lầy
Peter	Tin mách miệng

Mệnh đề WHERE

Mệnh đề WHERE được sử dụng để trích xuất chỉ những hồ sơ mà thực hiện một tiêu chuẩn quy định.

Cú pháp

```
SELECT column_name(s)
FROM table_name
WHERE column_name operator value
```

Để tìm hiểu thêm về SQL, vui lòng truy cập của chúng tôi [hướng dẫn SQL](#) .

Để có được PHP để thực hiện các tuyên bố trên, chúng ta phải sử dụng hàm `mysql_query()`. Chức năng này được sử dụng để gửi một truy vấn hoặc lệnh để kết nối MySQL.

Ví dụ

Ví dụ sau đây chọn tất cả các hàng từ bảng "Người" "FirstName = 'Peter'":

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

$result = mysql_query("SELECT * FROM Persons
WHERE FirstName='Peter'");

while($row = mysql_fetch_array($result))
{
    echo $row['FirstName'] . " " . $row['LastName'];
    echo "<br>";
}
?>
```

Đầu ra của các mã trên sẽ là:

```
Peter Griffin
```

ORDER BY Từ khóa

ORDER BY từ khóa được sử dụng để sắp xếp dữ liệu trong một recordset.

ORDER BY loại từ khóa các bản ghi trong thứ tự tăng dần theo mặc định.

Nếu bạn muốn sắp xếp các bản ghi trong một thứ tự giảm dần, bạn có thể sử dụng các từ khoá DESC.

Cú pháp

```
SELECT column_name(s)
FROM table_name
ORDER BY column_name(s) ASC|DESC
```

Để tìm hiểu thêm về SQL, vui lòng truy cập của chúng tôi [hướng dẫn SQL](#) .

Ví dụ

Ví dụ sau đây chọn tất cả các dữ liệu được lưu trữ trong bảng "Người", và sắp xếp kết quả theo cột "Age":

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

$result = mysql_query("SELECT * FROM Persons ORDER BY age");

while($row = mysql_fetch_array($result))
{
    echo $row['FirstName'];
    echo " " . $row['LastName'];
    echo " " . $row['Age'];
    echo "<br>";
}

mysql_close($con);
?>
```

Đầu ra của các mã trên sẽ là:

```
Glenn Quagmire 33
Peter Griffin 35
```

Sắp xếp theo hai cột

Nó cũng có thể để tự do nhiều hơn một cột. Khi đặt hàng theo nhiều hơn một cột, cột thứ hai chỉ được sử dụng nếu các giá trị trong cột đầu tiên là bằng nhau:

```
SELECT column_name(s)
FROM table_name
ORDER BY column1, column2
```

Cập nhật dữ liệu trong một cơ sở dữ liệu

Câu lệnh UPDATE được sử dụng để cập nhật các bản ghi hiện có trong một bảng.

Cú pháp

```
UPDATE table_name
SET column1=value, column2=value2,...
WHERE some_column=some_value
```

Lưu ý: Chú ý mệnh đề WHERE trong cú pháp UPDATE. Mệnh đề WHERE quy định cụ thể hồ sơ hoặc hồ sơ cần được cập nhật. Nếu bạn bỏ qua mệnh đề WHERE, tất cả hồ sơ sẽ được cập nhật!

Để tìm hiểu thêm về SQL, vui lòng truy cập của chúng tôi [hướng dẫn SQL](#) .

Để có được PHP để thực hiện các tuyên bố trên, chúng ta phải sử dụng hàm mysql_query (). Chức năng này được sử dụng để gửi một truy vấn hoặc lệnh để kết nối MySQL.

Ví dụ

Trước đó, trong hướng dẫn, chúng tôi tạo ra một bảng có tên là "Người". Đây là nó trông như thế nào:

FirstName	LastName	Tuổi
Peter	Tin mách miệng	35
Glenn	Bai lầy	33

Ví dụ cập nhật sau đây một số dữ liệu trong bảng "Persons":

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

mysql_query("UPDATE Persons SET Age=36
WHERE FirstName='Peter' AND LastName='Griffin'");

mysql_close($con);
?>
```

Sau khi cập nhật, bảng "Người" sẽ như thế này:

FirstName	LastName	Tuổi
Peter	Tin mách miệng	36
Glenn	Bai lầy	33

Xóa dữ liệu trong cơ sở dữ liệu

DELETE FROM tuyên bố được sử dụng để xóa các bản ghi từ một bảng cơ sở dữ liệu.

Cú pháp

```
DELETE FROM table_name
WHERE some_column = some_value
```

Lưu ý: Chú ý mệnh đề WHERE trong cú pháp DELETE. Mệnh đề WHERE quy định cụ thể hồ sơ hoặc hồ sơ có cần xóa. Nếu bạn bỏ qua mệnh đề WHERE, tất cả hồ sơ sẽ bị xóa!

Để tìm hiểu thêm về SQL, vui lòng truy cập của chúng tôi [hướng dẫn SQL](#) .

Để có được PHP để thực hiện các tuyên bố trên, chúng ta phải sử dụng hàm mysql_query (). Chức năng này được sử dụng để gửi một truy vấn hoặc lệnh để kết nối MySQL.

Ví dụ

Nhìn vào bảng sau "Người":

FirstName	LastName	Tuổi
Peter	Tin mách miệng	35
Glenn	Bai lầy	33

Các ví dụ sau đây xóa tất cả các bản ghi trong bảng "Persons" LastName = 'Griffin':

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

mysql_query("DELETE FROM Persons WHERE LastName='Griffin'");

mysql_close($con);
?>
```

Sau khi xóa, bảng sẽ như thế này:

FirstName	LastName	Tuổi
Glenn	Bai lầy	33

Tạo một kết nối ODBC

Với một kết nối ODBC, bạn có thể kết nối vào cơ sở dữ liệu nào, trên bất kỳ máy tính nào trong mạng của bạn, miễn là có một kết nối ODBC.

Đây là cách để tạo ra một kết nối ODBC một cơ sở dữ liệu MS Access:

1. Mở biểu tượng **Administrative Tools** trong Control Panel của bạn.
2. Double-click vào **Data Sources (ODBC)** biểu tượng bên trong.
3. Chọn tab **System DSN**.
4. Nhấp vào **Thêm** trong tab System DSN.
5. Chọn **Microsoft Access Driver**. Nhấp vào **Finish**.
6. Trong màn hình kế tiếp, nhấp vào **Chọn** để xác định vị trí các cơ sở dữ liệu.
7. Cung cấp cho các cơ sở dữ liệu **Data Source Name (DSN)**.
8. Nhấp vào **OK**.

Lưu ý rằng cấu hình này được thực hiện trên máy tính, nơi trang web của bạn nằm. Nếu bạn đang chạy Internet Information Server (IIS) trên máy tính của riêng của bạn, hướng dẫn ở trên sẽ làm việc, nhưng nếu trang web của bạn nằm trên một máy chủ từ xa, bạn phải có quyền truy cập vật lý vào máy chủ đó, hoặc yêu cầu máy chủ web của bạn để thiết lập một DSN cho bạn sử dụng.

Kết nối đến một ODBC

Odbc_connect () được sử dụng để kết nối với một nguồn dữ liệu ODBC. Chức năng có bốn thông số: nguồn dữ liệu tên, tên người dùng, mật khẩu, và con trỏ một kiểu tùy chọn.

Odbc_exec () chức năng được sử dụng để thực thi một câu lệnh SQL.

Ví dụ

Ví dụ sau đây tạo ra một kết nối đến một DSN gọi là northwind, không có tên người dùng và mật khẩu không. Sau đó tạo ra một câu lệnh SQL và thực hiện nó:

```
$conn=odbc_connect('northwind','','');
$sql="SELECT * FROM customers";
$rs=odbc_exec($conn,$sql);
```

Lấy hồ sơ

Chức năng odbc_fetch_row () được sử dụng để trả lại hồ sơ từ tập kết quả. Chức năng này trả về true nếu nó có thể trả lại hàng, nếu không sai.

Chức năng này có hai tham số: định danh kết quả ODBC và hàng một số tùy chọn:

```
odbc_fetch_row($rs)
```

Lấy Fields từ mức kỷ lục

Chức năng `odbc_result()` được sử dụng để đọc các lĩnh vực từ mức kỷ lục. Chức năng này có hai tham số: định danh kết quả ODBC và một số tên trường.

Các dòng mã dưới đây trả về giá trị của lĩnh vực đầu tiên từ các hồ sơ:

```
$compname=odbc_result($rs,1);
```

Các dòng mã dưới đây trả về giá trị của một lĩnh vực được gọi là "CompanyName":

```
$compname=odbc_result($rs,"CompanyName");
```

Đóng một kết nối ODBC

Chức năng `odbc_close()` được sử dụng để đóng một kết nối ODBC.

```
odbc_close($conn);
```

Ví dụ ODBC

Ví dụ sau đây cho thấy làm thế nào để lần đầu tiên tạo ra một kết nối cơ sở dữ liệu, sau đó thiết lập một kết quả, và sau đó hiển thị các dữ liệu trong một bảng HTML.

```
<html>
<body>

<?php
$conn=odbc_connect('northwind','','');
if (!$conn)
    {exit("Connection Failed: " . $conn);}
$sql="SELECT * FROM customers";
$rs=odbc_exec($conn,$sql);
if (!$rs)
    {exit("Error in SQL");}
echo "<table><tr>";
echo "<th>Companyname</th>";
echo "<th>Contactname</th></tr>";
while (odbc_fetch_row($rs))
    {
        $compname=odbc_result($rs,"CompanyName");
```

```
$conname=odbc_result($rs,"ContactName");  
echo "<tr><td>$conname</td>";  
echo "<td>$conname</td></tr>";  
}  
odbc_close($conn);  
echo "</table>";  
?>  
  
</body>  
</html>
```

Chapter 28 : PHP XML Expat Parser

Phân tích cú pháp nước ngoài được xây dựng trong làm cho nó có thể để xử lý các tài liệu XML trong PHP.

XML là gì?

XML được sử dụng để mô tả dữ liệu và tập trung vào dữ liệu là gì. Một tập tin XML mô tả cấu trúc của dữ liệu.

Trong XML, không có thể được định nghĩa trước. Bạn phải xác định các thẻ riêng của mình.

Nếu bạn muốn tìm hiểu thêm về XML, vui lòng truy cập của chúng tôi [hướng dẫn XML](#) .

Người nước ngoài là gì?

Đọc và cập nhật - tạo ra và thao tác một tài liệu XML, bạn sẽ cần một phân tích cú pháp XML.

Có hai loại cơ bản của phân tích cú pháp XML:

- Dựa trên cây phân tích cú pháp: phân tích cú pháp này biến đổi một tài liệu XML vào một cấu trúc cây. Nó phân tích toàn bộ tài liệu, và cung cấp truy cập đến các yếu tố cây. ví dụ như Document Object Model (DOM)
- Phân tích cú pháp dựa trên sự kiện: Xem một tài liệu XML như là một loạt các sự kiện. Khi một sự kiện cụ thể xảy ra, nó gọi một chức năng để xử lý nó

Các phân tích cú pháp nước ngoài là một bộ phân tích dựa trên sự kiện.

Dựa trên sự kiện phân tích cú pháp tập trung vào nội dung của các tài liệu XML, cấu trúc của chúng. Bởi vì điều này, phân tích cú pháp dựa trên sự kiện có thể truy cập dữ liệu nhanh hơn so với các phân tích cú pháp dựa trên cây.

Nhìn vào các phần XML sau đây:

```
<from>Jani</from>
```

Một phân tích cú pháp dựa trên sự kiện báo cáo XML ở trên là một loạt các ba sự kiện:

- Yếu tố: từ
- Bắt đầu CDATA phần giá trị: Jani
- Đóng yếu tố: từ

Ví dụ XML ở trên chứa tốt được hình thành XML. Tuy nhiên, ví dụ này là không hợp lệ XML, vì không có định nghĩa kiểu tài liệu (DTD) liên kết với nó.

Tuy nhiên, điều này làm cho không có sự khác biệt khi sử dụng các phân tích cú pháp nước ngoài. Người nước ngoài là một phân tích cú pháp không kiểm duyệt, và bỏ qua bất kỳ các DTD.

Là một sự kiện, không kiểm duyệt phân tích cú pháp XML, Người nước ngoài là nhanh chóng và nhỏ, và một trận đấu hoàn hảo cho các ứng dụng web PHP.

Lưu ý: các tài liệu XML phải được hình thành hoặc nước ngoài sẽ tạo ra một lỗi.

Cài đặt

Các chức năng phân tích cú pháp XML Expat là một phần của lõi PHP. Không có cài đặt cần thiết để sử dụng các chức năng này.

Một file XML

Các tập tin XML dưới đây sẽ được sử dụng trong ví dụ của chúng tôi:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```

Khởi tạo phân tích cú pháp XML

Chúng tôi muốn để khởi tạo phân tích cú pháp XML trong PHP, xác định một số xử lý cho các sự kiện khác nhau XML, và sau đó phân tích các tập tin XML.

Ví dụ

```
<?php
//Initialize the XML parser
$parser=xml_parser_create();

//Function to use at the start of an element
function start($parser,$element_name,$element_attrs)
{
    switch($element_name)
    {
        case "NOTE":
            echo "-- Note --<br>";
            break;
        case "TO":
            echo "To: ";
            break;
        case "FROM":
            echo "From: ";
            break;
        case "HEADING":
            echo "Heading: ";
            break;
        case "BODY":
            echo "Message: ";
            break;
    }
}

//Function to use at the end of an element
function stop($parser,$element_name)
{
    echo "<br>";
}

//Function to use when finding character data
function char($parser,$data)
{
    echo $data;
}

//Specify element handler
xml_set_element_handler($parser,"start","stop");

//Specify data handler
xml_set_character_data_handler($parser,"char");

//Open XML file
$fp=fopen("test.xml","r");

//Read data
while ($data=fread($fp,4096))
{
    xml_parse($parser,$data,feof($fp)) or
    die (sprintf("XML Error: %s at line %d",
```

```
xml_error_string(xml_get_error_code($parser)),
xml_get_current_line_number($parser));
}

//Free the XML parser
xml_parser_free($parser);
?>
```

Đầu ra của các mã trên sẽ là:

```
-- Note --
To: Tove
From: Jani
Heading: Reminder
Message: Don't forget me this weekend!
```

Làm thế nào nó hoạt động:

1. Khởi tạo phân tích cú pháp XML với `xml_parser_create()` chức năng
2. Tạo các chức năng để sử dụng với các xử lý sự kiện khác nhau
3. Thêm `xml_set_element_handler()` chức năng để xác định có chức năng sẽ được thực hiện khi phân tích cú pháp gặp các thẻ mở và đóng
4. Thêm `xml_set_character_data_handler()` chức năng để xác định các chức năng sẽ thực hiện khi phân tích cú pháp gặp dữ liệu ký tự
5. Phân tích các tập tin "test.xml" với `xml_parse()` chức năng
6. Trong trường hợp có lỗi, thêm `xml_error_string()` chức năng để chuyển đổi một lỗi XML với một mô tả văn bản
7. Gọi `xml_parser_free()` chức năng để giải phóng bộ nhớ được cấp phát bằng `xml_parser_create()` chức năng

Thêm nước ngoài PHP Parser

Để biết thêm thông tin về các chức năng nước ngoài PHP, hãy truy cập của chúng tôi [tham khảo PHP Parser XML](#).

Việc xây dựng trong DOM phân tích cú pháp làm cho nó có thể để xử lý các tài liệu XML trong PHP.

DOM là gì?

W3C DOM cung cấp một bộ tiêu chuẩn của các đối tượng cho tài liệu HTML và XML, và một giao diện tiêu chuẩn cho việc truy cập và thao tác chúng. W3C DOM được chia thành các phần khác nhau (Core, XML, HTML) và các mức độ khác nhau (DOM Level 1/2 / 3): * Core DOM - định nghĩa một tập hợp các đối tượng cho bất kỳ tài liệu có cấu trúc * XML DOM - định nghĩa một bộ tiêu chuẩn của các đối tượng cho các tài liệu XML * HTML DOM - định nghĩa một bộ tiêu chuẩn của các đối tượng cho các tài liệu HTML Nếu bạn muốn tìm hiểu thêm về DOM XML, vui lòng truy cập của chúng tôi [hướng dẫn XML DOM](#).

XML Parsing

Đọc và cập nhật - tạo ra và thao tác một tài liệu XML, bạn sẽ cần một phân tích cú pháp XML.

Có hai loại cơ bản của phân tích cú pháp XML:

- Dựa trên cây phân tích cú pháp: phân tích cú pháp này biến đổi một tài liệu XML vào một cấu trúc cây. Nó phân tích toàn bộ tài liệu, và cung cấp truy cập đến các yếu tố cây
- Phân tích cú pháp dựa trên sự kiện: Xem một tài liệu XML như là một loạt các sự kiện. Khi một sự kiện cụ thể xảy ra, nó gọi một chức năng để xử lý nó

Các phân tích cú pháp DOM là một phân tích cú pháp dựa trên cây.

Nhìn vào phần tài liệu XML sau:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<from>Jani</from>
```

XML DOM XML nhìn thấy ở trên là một cấu trúc cây:

- Level 1: XML Document
- Level 2: Root yếu tố: <from>
- Level 3: Tiêu đề yếu tố: "Jani"

Cài đặt

DOM XML chức năng phân tích cú pháp là một phần của lõi PHP. Không có cài đặt cần thiết để sử dụng các chức năng này.

Một file XML

Các tập tin XML dưới đây sẽ được sử dụng trong ví dụ của chúng tôi:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```

Tải và đầu ra XML

Chúng tôi muốn để khởi tạo phân tích cú pháp XML, tải xml, và đầu ra nó:

Ví dụ

```
<?php
$xmlDoc = new DOMDocument();
$xmlDoc->load("note.xml");

print $xmlDoc->saveXML();
?>
```

Đầu ra của các mã trên sẽ là:

```
Tove Jani Reminder Don't forget me this weekend!
```

Nếu bạn chọn "Xem mã nguồn" trong cửa sổ trình duyệt, bạn sẽ thấy mã HTML sau:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```

Ví dụ trên tạo ra một DOMDocument đối tượng và tải XML từ "note.xml" vào nó.

Sau đó, saveXML () chức năng đặt các tài liệu nội bộ XML thành một chuỗi, vì vậy chúng tôi có thể sản xuất nó.

Looping thông qua XML

Chúng tôi muốn khởi tạo phân tích cú pháp XML, nạp XML, và vòng lặp thông qua tất cả các yếu tố của phần tử <note>:

Ví dụ

```
<?php
$xmlDoc = new DOMDocument();
```

```
$xmlDoc->load("note.xml");

$x = $xmlDoc->documentElement;
foreach ($x->childNodes AS $item)
{
    print $item->nodeName . " = " . $item->nodeValue . "<br>";
}
?>
```

Đầu ra của các mã trên sẽ là:

```
#text =
to = Tove
#text =
from = Jani
#text =
heading = Reminder
#text =
body = Don't forget me this weekend!
#text =
```

Trong ví dụ trên, bạn thấy rằng có các nút văn bản trống giữa mỗi phần tử.

Khi XML tạo ra, nó thường có màu trắng không gian giữa các nút. XML DOM phân tích cú pháp xử lý các yếu tố này là người bình thường, và nếu bạn không nhận thức của họ, đôi khi họ gây ra vấn đề.

Nếu bạn muốn tìm hiểu thêm về XML DOM, vui lòng truy cập của chúng tôi [hướng dẫn XML DOM](#) .

SimpleXML là gì?

SimpleXML là mới trong PHP 5. Đó là một cách dễ dàng nhận được các thuộc tính của một nguyên tố và văn bản, nếu bạn biết bố trí các tài liệu XML.

So với DOM hoặc phân tích cú pháp nước ngoài, SimpleXML chỉ mất một vài dòng mã để đọc dữ liệu văn bản từ một phần tử.

SimpleXML chuyển đổi các tài liệu XML vào một đối tượng, như thế này:

- Elements - chuyển đổi các thuộc tính duy nhất của các đối tượng SimpleXMLElement. Khi có nhiều hơn một phần tử ở một mức độ, họ đang đặt bên trong một mảng
- Thuộc tính - truy cập bằng cách sử dụng mảng kết hợp, nơi mà chỉ mục tương ứng với tên thuộc tính
- Yếu tố dữ liệu - dữ liệu văn bản từ các yếu tố được chuyển đổi thành chuỗi. Nếu một phần tử có nhiều hơn một nút văn bản, họ sẽ được sắp xếp theo thứ tự chúng được tìm thấy

SimpleXML là nhanh chóng và dễ dàng để sử dụng khi thực hiện nhiệm vụ cơ bản như:

- Đọc các tập tin XML
- Nén dữ liệu từ chuỗi XML
- Chỉnh sửa các nút văn bản hoặc các thuộc tính

Tuy nhiên, khi đối phó với XML tiên tiến, như không gian tên, bạn tốt hơn bằng cách sử dụng các phân tích cú pháp nước ngoài hoặc DOM XML.

Cài đặt

PHP 5.0, các chức năng SimpleXML là một phần của lõi PHP. Không có cài đặt cần thiết để sử dụng các chức năng này.

Sử dụng SimpleXML

Dưới đây là một tập tin XML:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```

Chúng tôi muốn sản xuất các tên phần tử và dữ liệu từ tập tin XML ở trên.

Dưới đây là những việc cần làm:

1. Tải các tập tin XML
2. Lấy tên của các yếu tố đầu tiên
3. Tạo một vòng lặp mà sẽ kích hoạt trên mỗi nút con, sử dụng con cái () chức năng
4. Sản xuất các tên phần tử và dữ liệu cho mỗi nút con

Ví dụ

```
<?php
$xml = simplexml_load_file("test.xml");

echo $xml->getName() . "<br>";

foreach($xml->children() as $child)
{
    echo $child->getName() . ": " . $child . "<br>";
}
?>
```

Đầu ra của các mã trên sẽ là:

```
note
to: Tove
from: Jani
heading: Reminder
body: Don't forget me this weekend!
```

More PHP SimpleXML

Để biết thêm thông tin về các chức năng PHP SimpleXML, hãy truy cập của chúng tôi [tham khảo PHP SimpleXML](#).

Chapter 29 : Giới thiệu AJAX

AJAX là về việc cập nhật các bộ phận của một trang web, mà không cần tải lại toàn bộ trang.

AJAX là gì?

AJAX = Asynchronous JavaScript và XML.

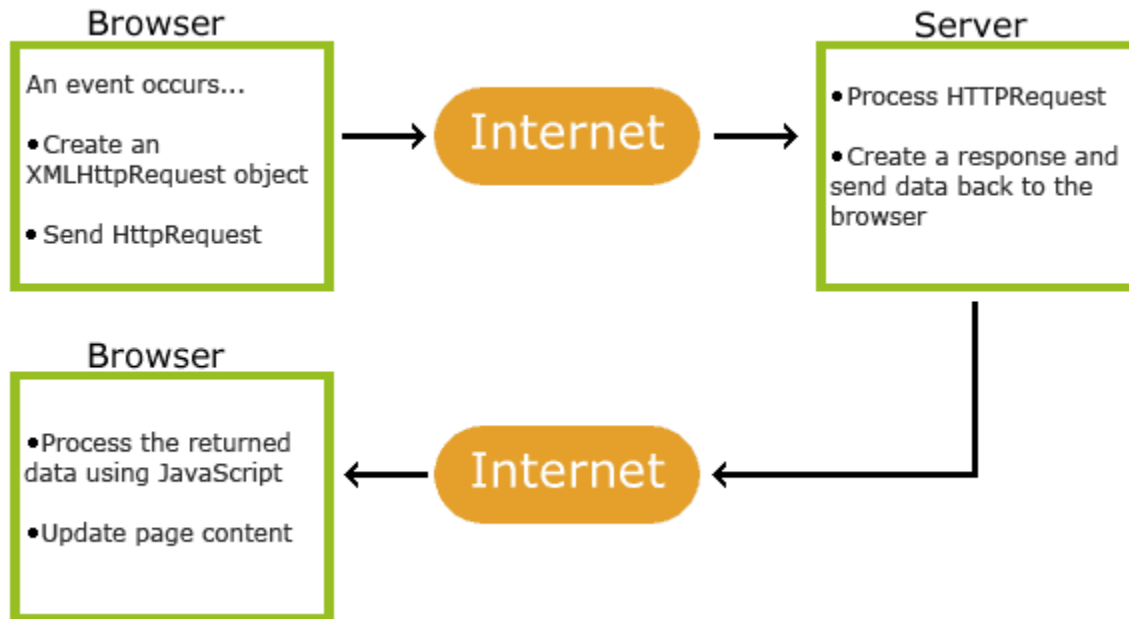
AJAX là một kỹ thuật để tạo ra các trang web nhanh chóng và năng động.

AJAX cho phép các trang web được cập nhật không đồng bộ bằng cách trao đổi một lượng nhỏ dữ liệu với máy chủ đằng sau hậu trường. Điều này có nghĩa rằng nó có thể cập nhật các phần của một trang web, mà không cần tải lại toàn bộ trang.

Các trang web cổ điển, (mà không sử dụng AJAX) phải tải lại toàn bộ trang nếu các nội dung cần thay đổi.

Ví dụ về các ứng dụng bằng cách sử dụng AJAX: Google Maps, Gmail, Youtube, và Facebook tab.

Làm thế nào AJAX Hoạt động



AJAX Dựa trên các tiêu chuẩn Internet

AJAX dựa trên các tiêu chuẩn internet, và sử dụng một sự kết hợp của:

- Đối tượng XMLHttpRequest (trao đổi dữ liệu không đồng bộ với một máy chủ)
- JavaScript / DOM (để hiển thị / tương tác với thông tin)
- CSS (kiểu dữ liệu)
- XML (thường được sử dụng như là định dạng để chuyển dữ liệu)



Các ứng dụng AJAX là trình duyệt và nền tảng độc lập!

Google Suggest

AJAX đã được thực hiện phổ biến vào năm 2005 bởi Google, với Google Suggest.

Google Suggest đang sử dụng AJAX để tạo ra một giao diện web rất năng động: Khi bạn bắt đầu gõ vào hộp tìm kiếm của Google, JavaScript một gửi các ký tự vào một máy chủ và máy chủ trả về một danh sách gợi ý.

Bắt đầu Sử dụng AJAX Hôm nay

Trong hướng dẫn PHP của chúng tôi, chúng tôi sẽ chứng minh làm thế nào AJAX có thể cập nhật các phần của một trang web, mà không cần tải lại toàn bộ trang. Các kịch bản máy chủ sẽ được viết bằng PHP.

Nếu bạn muốn tìm hiểu thêm về AJAX, hãy truy cập của chúng tôi [hướng dẫn AJAX](#) .

AJAX PHP Ví dụ

Ví dụ sau đây sẽ chứng minh làm thế nào một trang web có thể giao tiếp với một máy chủ web trong khi một người sử dụng loại nhân vật trong một lĩnh vực đầu vào:

Ví dụ

Start typing a name in the input field below:

First name:

Suggestions:



Ví dụ Giải thích - Trang HTML

Khi người dùng gõ một ký tự trong lĩnh vực đầu vào ở trên, chức năng "showHint ()" được thực hiện. Chức năng này được kích hoạt bởi sự kiện "onkeyup":

```
<html>
<head>
<script>
function showHint(str)
{
if (str.length==0)
{
document.getElementById("txtHint").innerHTML="";
return;
}
if (window.XMLHttpRequest)
{// code for IE7+, Firefox, Chrome, Opera, Safari
xmlhttp=new XMLHttpRequest();
}
else
{// code for IE6, IE5
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange=function()
{
if (xmlhttp.readyState==4 && xmlhttp.status==200)
{
document.getElementById("txtHint").innerHTML=xmlhttp.responseText;
```

```

    }
}
xmlhttp.open("GET", "gethint.php?q="+str,true);
xmlhttp.send();
}
</script>
</head>
<body>

<p><b>Start typing a name in the input field below:</b></p>
<form>
First name: <input type="text" onkeyup="showHint(this.value)">
</form>
<p>Suggestions: <span id="txtHint"></span></p>

</body>
</html>

```

Mã nguồn giải thích:

Nếu trường đầu vào là rỗng (`str.length == 0`), chức năng xóa nội dung giữ chỗ `txtHint` ra khỏi chức năng.

Nếu trường đầu vào không phải là trống rỗng, `showHint()` thực hiện các chức năng sau đây:

- Tạo một đối tượng `XMLHttpRequest`
- Tạo ra các chức năng được thực hiện khi đáp ứng máy chủ đã sẵn sàng
- Gửi yêu cầu vào một tập tin trên máy chủ
- Chú ý rằng một tham số (`q`) được thêm vào URL (với nội dung của lĩnh vực đầu vào)

File PHP

Các trang web trên máy chủ được gọi bởi JavaScript ở trên là một file PHP gọi là "gethint.php".

Mã nguồn trong "gethint.php" kiểm tra một loạt các tên, và trả về tên tương ứng (s) cho trình duyệt:

```

<?php
// Fill up array with names
$a[]="Anna";
$a[]="Brittany";
$a[]="Cinderella";
$a[]="Diana";
$a[]="Eva";
$a[]="Fiona";
$a[]="Gunda";
$a[]="Hege";
$a[]="Inga";
$a[]="Johanna";
$a[]="Kitty";
$a[]="Linda";
$a[]="Nina";

```

```

$a[]="Ophelia";
$a[]="Petunia";
$a[]="Amanda";
$a[]="Raquel";
$a[]="Cindy";
$a[]="Doris";
$a[]="Eve";
$a[]="Evita";
$a[]="Sunniva";
$a[]="Tove";
$a[]="Unni";
$a[]="Violet";
$a[]="Liza";
$a[]="Elizabeth";
$a[]="Ellen";
$a[]="Wenche";
$a[]="Vicky";

//get the q parameter from URL
$q=$_GET["q"];

//lookup all hints from array if length of q>0
if (strlen($q) > 0)
{
    $hint="";
    for($i=0; $i<count($a); $i++)
    {
        if (strtolower($q)==strtolower(substr($a[$i],0,strlen($q)))
        {
            if ($hint=="")
            {
                $hint=$a[$i];
            }
            else
            {
                $hint=$hint." , ".$a[$i];
            }
        }
    }
}

// Set output to "no suggestion" if no hint were found
// or to the correct values
if ($hint == "")
{
    $response="no suggestion";
}
else
{
    $response=$hint;
}

//output the response
echo $response;
?>

```

Giải thích: Nếu có bất kỳ văn bản được gửi từ JavaScript (`strlen ($ q) > 0`), sau đây xảy ra:

1. Tìm một cái tên phù hợp với các ký tự được gửi từ JavaScript
2. Nếu không phù hợp được tìm thấy, thiết lập các chuỗi phản ứng "đề nghị không"
3. Nếu một hoặc nhiều tên phù hợp được tìm thấy, thiết lập các chuỗi phản ứng cho tất cả các tên này
4. Đáp ứng được gửi đến giữ chỗ "txtHint"

AJAX Cơ sở dữ liệu Ví dụ

Ví dụ sau đây sẽ chứng minh làm thế nào một trang web có thể lấy thông tin từ một cơ sở dữ liệu với AJAX:

Ví dụ

Person info will be listed here...

Ví dụ Giải thích - Cơ sở dữ liệu MySQL

Bảng cơ sở dữ liệu chúng tôi sử dụng trong ví dụ trên trông như thế này:

id	FirstName	LastName	Tuổi	Quê hương	Việc
1	Peter	Tin mạch miệng	41	Quahog	Chỗ làm rượu bia
2	Lois	Tin mạch miệng	40	Newport	Piano Teacher
3	Người đàn ông chay tịnh	Swanson	39	Quahog	Cảnh sát viên
4	Glenn	Bai lầy	41	Quahog	Phi công

Ví dụ Giải thích - Trang HTML

Khi người dùng chọn một người sử dụng trong danh sách thả xuống ở trên, một chức năng được gọi là "showUser ()" được thực thi. Chức năng này được kích hoạt bởi sự kiện "onchange":

```
<html>
<head>
<script>
function showUser(str)
{
  if (str=="")
  {
    document.getElementById("txtHint").innerHTML="";
  }
}
```

```

    return;
}
if (window.XMLHttpRequest)
    { // code for IE7+, Firefox, Chrome, Opera, Safari
    xmlhttp=new XMLHttpRequest();
    }
else
    { // code for IE6, IE5
    xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
    }
xmlhttp.onreadystatechange=function()
{
    if (xmlhttp.readyState==4 && xmlhttp.status==200)
    {
        document.getElementById("txtHint").innerHTML=xmlhttp.responseText;
    }
}
xmlhttp.open("GET","getuser.php?q="+str,true);
xmlhttp.send();
}
</script>
</head>
<body>

<form>
<select name="users" onchange="showUser(this.value)">
<option value="">Select a person:</option>
<option value="1">Peter Griffin</option>
<option value="2">Lois Griffin</option>
<option value="3">Glenn Quagmire</option>
<option value="4">Joseph Swanson</option>
</select>
</form>
<br>
<div id="txtHint"><b>Person info will be listed here.</b></div>

</body>
</html>

```

ShowUser () chức năng nào sau đây:

- Kiểm tra nếu một người được chọn
- Tạo một đối tượng XMLHttpRequest
- Tạo ra các chức năng được thực hiện khi đáp ứng máy chủ đã sẵn sàng
- Gửi yêu cầu vào một tập tin trên máy chủ
- Chú ý rằng một tham số (q) được thêm vào URL (với nội dung của danh sách thả xuống)

File PHP

Các trang web trên máy chủ được gọi bởi JavaScript ở trên là một file PHP gọi là "getuser.php".

Mã nguồn trong "getuser.php" chạy một truy vấn đối với một cơ sở dữ liệu MySQL, và trả về kết quả trong một bảng HTML:

```
<?php
$q=$_GET["q"];

$con = mysql_connect('localhost', 'peter', 'abc123');
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("ajax_demo", $con);

$sql="SELECT * FROM user WHERE id = '". $q. "'";

$result = mysql_query($sql);

echo "<table border='1'>
<tr>
<th>Firstname</th>
<th>Lastname</th>
<th>Age</th>
<th>Hometown</th>
<th>Job</th>
</tr>";

while($row = mysql_fetch_array($result))
{
    echo "<tr>";
    echo "<td>" . $row['FirstName'] . "</td>";
    echo "<td>" . $row['LastName'] . "</td>";
    echo "<td>" . $row['Age'] . "</td>";
    echo "<td>" . $row['Hometown'] . "</td>";
    echo "<td>" . $row['Job'] . "</td>";
    echo "</tr>";
}
echo "</table>";

mysql_close($con);
?>
```

Giải thích: Khi truy vấn được gửi từ JavaScript để file PHP, sau đây xảy ra:

1. PHP mở một kết nối đến một máy chủ MySQL
2. Người được tìm thấy đúng
3. Một bảng HTML được tạo ra, chứa đầy dữ liệu, và gửi lại cho giữ chỗ "txtHint"

AJAX XML Ví dụ

Ví dụ sau đây sẽ chứng minh làm thế nào một trang web có thể lấy thông tin từ một file XML với AJAX:

Ví dụ

CD info will be listed here...

Ví dụ Giải thích - Trang HTML

Khi người dùng chọn một đĩa CD trong danh sách thả xuống ở trên, một chức năng được gọi là "showCD ()" được thực thi. Chức năng này được kích hoạt bởi sự kiện "onchange":

```
<html>
<head>
<script>
function showCD(str)
{
if (str=="")
{
document.getElementById("txtHint").innerHTML="";
return;
}
if (window.XMLHttpRequest)
{// code for IE7+, Firefox, Chrome, Opera, Safari
xmlhttp=new XMLHttpRequest();
}
else
{// code for IE6, IE5
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange=function()
{
if (xmlhttp.readyState==4 && xmlhttp.status==200)
{
document.getElementById("txtHint").innerHTML=xmlhttp.responseText;
}
}
xmlhttp.open("GET","getcd.php?q="+str,true);
xmlhttp.send();
}
</script>
</head>
<body>

<form>
Select a CD:
<select name="cds" onchange="showCD(this.value)">
<option value="">Select a CD:</option>
<option value="Bob Dylan">Bob Dylan</option>
<option value="Bonnie Tyler">Bonnie Tyler</option>
<option value="Dolly Parton">Dolly Parton</option>
</select>
</form>
<div id="txtHint"><b>CD info will be listed here...</b></div>
```

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

ShowCD () chức năng nào sau đây:

- Kiểm tra xem nếu đĩa CD được chọn
- Tạo một đối tượng XMLHttpRequest
- Tạo ra các chức năng được thực hiện khi đáp ứng máy chủ đã sẵn sàng
- Gửi yêu cầu vào một tập tin trên máy chủ
- Chú ý rằng một tham số (q) được thêm vào URL (với nội dung của danh sách thả xuống)

File PHP

Các trang web trên máy chủ được gọi bởi JavaScript ở trên là một file PHP gọi là "getcd.php".

Các kịch bản PHP tải một tài liệu XML, "cd_catalog.xml", chạy một truy vấn đối với các tập tin XML, và trả về kết quả dưới dạng HTML:

```
<?php
$q=$_GET["q"];

$xmlDoc = new DOMDocument();
$xmlDoc->load("cd_catalog.xml");

$x=$xmlDoc->getElementsByTagName('ARTIST');

for ($i=0; $i<=$x->length-1; $i++)
{
    //Process only element nodes
    if ($x->item($i)->nodeType==1)
    {
        if ($x->item($i)->childNodes->item(0)->nodeValue == $q)
        {
            $y=($x->item($i)->parentNode);
        }
    }
}

$cd=($y->childNodes);

for ($i=0;$i<$cd->length;$i++)
{
    //Process only element nodes
    if ($cd->item($i)->nodeType==1)
    {
        echo("<b>" . $cd->item($i)->nodeName . ":</b> ");
        echo($cd->item($i)->childNodes->item(0)->nodeValue);
        echo("<br>");
    }
}
?>
```

Khi truy vấn CD được gửi từ JavaScript trang PHP, sau đây sẽ xảy ra:

1. PHP tạo ra một đối tượng XML DOM
2. Tìm tất cả các yếu tố <artist> phù hợp với tên được gửi từ JavaScript
3. Đầu ra các thông tin album (gửi cho giữ chỗ "txtHint")

AJAX Live Search

Ví dụ sau đây sẽ chứng minh một tìm kiếm trực tiếp, nơi bạn sẽ có được kết quả tìm kiếm trong khi bạn gõ.

Live tìm kiếm có nhiều lợi ích so với tìm kiếm truyền thống:

- Kết quả được hiển thị khi bạn gõ
- Thu hẹp kết quả như bạn tiếp tục gõ
- Nếu kết quả trở nên quá hẹp, loại bỏ các ký tự nhìn thấy một kết quả rộng lớn hơn

Tìm kiếm một trang W3Schools trong lĩnh vực đầu vào dưới đây:

Các kết quả trong ví dụ trên được tìm thấy trong một tập tin XML ([links.xml](#)). Để làm ví dụ này nhỏ và đơn giản, chỉ có sáu kết quả có sẵn.

Ví dụ Giải thích - Trang HTML

Khi người dùng gõ một ký tự trong lĩnh vực đầu vào ở trên, chức năng "showResult ()" được thực hiện. Chức năng này được kích hoạt bởi sự kiện "onkeyup":

```
<html>
<head>
<script>
function showResult(str)
{
if (str.length==0)
{
document.getElementById("livesearch").innerHTML="";
document.getElementById("livesearch").style.border="0px";
return;
}
if (window.XMLHttpRequest)
{ // code for IE7+, Firefox, Chrome, Opera, Safari
xmlhttp=new XMLHttpRequest();
}
else
{ // code for IE6, IE5
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange=function()
```

```

{
  if (xmlhttp.readyState==4 && xmlhttp.status==200)
  {

    document.getElementById("livesearch").innerHTML=xmlhttp.responseText;
    document.getElementById("livesearch").style.border="1px solid
#A5ACB2";
  }
}
xmlhttp.open("GET","livesearch.php?q="+str,true);
xmlhttp.send();
}
</script>
</head>
<body>

<form>
<input type="text" size="30" onkeyup="showResult(this.value)">
<div id="livesearch"></div>
</form>

</body>
</html>

```

Mã nguồn giải thích:

Nếu trường đầu vào là rỗng (str.length == 0), chức năng xóa nội dung giữ chỗ livesearch và thoát khỏi chức năng.

Nếu trường đầu vào không phải là trống rỗng, showResult () chức năng thực hiện những điều sau đây:

- Tạo một đối tượng XMLHttpRequest
- Tạo ra các chức năng được thực hiện khi đáp ứng máy chủ đã sẵn sàng
- Gửi yêu cầu vào một tập tin trên máy chủ
- Chú ý rằng một tham số (q) được thêm vào URL (với nội dung của lĩnh vực đầu vào)

File PHP

Các trang web trên máy chủ được gọi bởi JavaScript ở trên là một file PHP gọi là "livesearch.php".

Mã nguồn trong "livesearch.php" tìm kiếm một tập tin XML cho các chức danh phù hợp với chuỗi tìm kiếm và trả về kết quả:

```

<?php
$xmlDoc=new DOMDocument();
$xmlDoc->load("links.xml");

$x=$xmlDoc->getElementsByTagName('link');

```

```

//get the q parameter from URL
$q=$_GET["q"];

//lookup all links from the xml file if length of q>0
if (strlen($q)>0)
{
    $hint="";
    for($i=0; $i<($x->length); $i++)
    {
        $y=$x->item($i)->getElementsByTagName('title');
        $z=$x->item($i)->getElementsByTagName('url');
        if ($y->item(0)->nodeType==1)
        {
            //find a link matching the search text
            if (stristr($y->item(0)->childNodes->item(0)->nodeValue,$q))
            {
                if ($hint=="")
                {
                    $hint="<a href='" .
                    $z->item(0)->childNodes->item(0)->nodeValue .
                    "' target='_blank'>" .
                    $y->item(0)->childNodes->item(0)->nodeValue . "</a>";
                }
                else
                {
                    $hint=$hint . "<br /><a href='" .
                    $z->item(0)->childNodes->item(0)->nodeValue .
                    "' target='_blank'>" .
                    $y->item(0)->childNodes->item(0)->nodeValue . "</a>";
                }
            }
        }
    }
}

// Set output to "no suggestion" if no hint were found
// or to the correct values
if ($hint=="")
{
    $response="no suggestion";
}
else
{
    $response=$hint;
}

//output the response
echo $response;
?>

```

Nếu có bất kỳ văn bản được gửi từ JavaScript (strlen (\$ q) > 0), sau đây sẽ xảy ra:

- Tải một tập tin XML vào một đối tượng mới DOM XML
- Lặp qua tất cả các yếu tố <title> tìm thấy kết quả phù hợp từ văn bản được gửi từ JavaScript

- Thiết lập địa chỉ chính xác và tiêu đề trong "phản ứng" biến. Nếu trận đấu nhiều hơn một được tìm thấy, tất cả các trận đấu được thêm vào biến
- Nếu không phù hợp được tìm thấy, biến phản ứng được thiết lập để "đề nghị không"

AJAX RSS Reader

Ví dụ sau đây sẽ chứng minh một trình đọc RSS, RSS-nguồn cấp dữ liệu được nạp vào một trang web mà không cần nạp:

RSS-nguồn cấp dữ liệu sẽ được liệt kê ở đây ...

Ví dụ Giải thích - Trang HTML

Khi người dùng chọn một RSS-nguồn cấp dữ liệu trong danh sách thả xuống ở trên, một chức năng được gọi là "showResult ()" được thực thi. Chức năng này được kích hoạt bởi sự kiện "onchange":

```
<html>
<head>
<script>
function showRSS(str)
{
if (str.length==0)
{
document.getElementById("rssOutput").innerHTML="";
return;
}
if (window.XMLHttpRequest)
{// code for IE7+, Firefox, Chrome, Opera, Safari
xmlhttp=new XMLHttpRequest();
}
else
{// code for IE6, IE5
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange=function()
{
if (xmlhttp.readyState==4 && xmlhttp.status==200)
{
document.getElementById("rssOutput").innerHTML=xmlhttp.responseText
;
}
}
xmlhttp.open("GET","getrss.php?q="+str,true);
xmlhttp.send();
}
</script>
</head>
<body>
```

```

<form>
<select onchange="showRSS(this.value)">
<option value="">Select an RSS-feed:</option>
<option value="Google">Google News</option>
<option value="MSNBC">MSNBC News</option>
</select>
</form>
<br>
<div id="rssOutput">RSS-feed will be listed here...</div>
</body>
</html>

```

Các showResult () chức năng nào sau đây:

- Kiểm tra nếu một RSS-nguồn cấp dữ liệu được chọn
- Tạo một đối tượng XMLHttpRequest
- Tạo ra các chức năng được thực hiện khi đáp ứng máy chủ đã sẵn sàng
- Gửi yêu cầu vào một tập tin trên máy chủ
- Chú ý rằng một tham số (q) được thêm vào URL (với nội dung của danh sách thả xuống)

File PHP

Các trang web trên máy chủ được gọi bởi JavaScript ở trên là một tập tin PHP gọi là "getrss.php":

```

<?php
//get the q parameter from URL
$q=$_GET["q"];

//find out which feed was selected
if($q=="Google")
{
$xml=("http://news.google.com/news?ned=us&topic=h&output=rss");
}
elseif($q=="MSNBC")
{
$xml=("http://rss.msnbc.msn.com/id/3032091/device/rss/rss.xml");
}

$xmlDoc = new DOMDocument();
$xmlDoc->load($xml);

//get elements from "<channel>"
$channel=$xmlDoc->getElementsByTagName('channel')->item(0);
$channel_title = $channel->getElementsByTagName('title')
->item(0)->childNodes->item(0)->nodeValue;
$channel_link = $channel->getElementsByTagName('link')
->item(0)->childNodes->item(0)->nodeValue;
$channel_desc = $channel->getElementsByTagName('description')
->item(0)->childNodes->item(0)->nodeValue;

```

```
//output elements from "<channel>"
echo("<p><a href='" . $channel_link
. "'>" . $channel_title . "</a>");
echo("<br>");
echo($channel_desc . "</p>");

//get and output "<item>" elements
$x=$xmlDoc->getElementsByTagName('item');
for ($i=0; $i<=2; $i++)
{
    $item_title=$x->item($i)->getElementsByTagName('title')
->item(0)->childNodes->item(0)->nodeValue;
    $item_link=$x->item($i)->getElementsByTagName('link')
->item(0)->childNodes->item(0)->nodeValue;
    $item_desc=$x->item($i)->getElementsByTagName('description')
->item(0)->childNodes->item(0)->nodeValue;

    echo ("<p><a href='" . $item_link
. "'>" . $item_title . "</a>");
    echo ("<br>");
    echo ($item_desc . "</p>");
}
?>
```

Khi một yêu cầu cho một nguồn cấp dữ liệu RSS được gửi từ các JavaScript, sau đây sẽ xảy ra:

- Kiểm tra thức ăn chăn nuôi đã được lựa chọn
- Tạo một đối tượng DOM XML mới
- Nạp tài liệu RSS trong biến xml
- Trích xuất và các yếu tố đầu ra từ các yếu tố kênh
- Trích xuất và các yếu tố đầu ra từ các yếu tố mục

Ví dụ sau đây sẽ chứng minh một cuộc thăm dò mà kết quả được hiển thị mà không cần nạp.

Bạn có thích PHP và AJAX cho đến nay?

Có: ☐

No: ☐

Ví dụ Giải thích - Trang HTML

Khi người dùng chọn một tùy chọn ở trên, một chức năng được gọi là "getVote ()" được thực thi. Chức năng này được kích hoạt bởi sự kiện "onclick":

```
<html>
<head>
<script>
function getVote(int)
{
    if (window.XMLHttpRequest)
```

```

    { // code for IE7+, Firefox, Chrome, Opera, Safari
      xmlhttp=new XMLHttpRequest();
    }
    else
    { // code for IE6, IE5
      xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
    }
    xmlhttp.onreadystatechange=function()
    {
      if (xmlhttp.readyState==4 && xmlhttp.status==200)
      {
        document.getElementById("poll").innerHTML=xmlhttp.responseText;
      }
    }
    xmlhttp.open("GET","poll_vote.php?vote="+int,true);
    xmlhttp.send();
  }
</script>
</head>
<body>

<div id="poll">
<h3>Do you like PHP and AJAX so far?</h3>
<form>
Yes:
<input type="radio" name="vote" value="0"
onclick="getVote(this.value)">
<br>No:
<input type="radio" name="vote" value="1"
onclick="getVote(this.value)">
</form>
</div>

</body>
</html>

```

GetVote () chức năng nào sau đây:

- Tạo một đối tượng XMLHttpRequest
- Tạo ra các chức năng được thực hiện khi đáp ứng máy chủ đã sẵn sàng
- Gửi yêu cầu vào một tập tin trên máy chủ
- Chú ý rằng một tham số (vote) được thêm vào URL (với giá trị có hoặc không có tùy chọn)

File PHP

Các trang web trên máy chủ được gọi bởi JavaScript ở trên là một tập tin PHP gọi là "poll_vote.php":

```

<?php
$vote = $_REQUEST['vote'];

```

```

//get content of textfile
$filename = "poll_result.txt";
$content = file($filename);

//put content in array
$array = explode("||", $content[0]);
$yes = $array[0];
$no = $array[1];

if ($vote == 0)
{
    $yes = $yes + 1;
}
if ($vote == 1)
{
    $no = $no + 1;
}

//insert votes to txt file
$insertvote = $yes."||".$no;
$fp = fopen($filename,"w");
fputs($fp,$insertvote);
fclose($fp);
?>

<h2>Result:</h2>
<table>
<tr>
<td>Yes:</td>
<td>
'
height='20'>
<?php echo(100*round($yes/($no+$yes),2)); ?>%
</td>
</tr>
<tr>
<td>No:</td>
<td>
'
height='20'>
<?php echo(100*round($no/($no+$yes),2)); ?>%
</td>
</tr>
</table>

```

Giá trị được gửi từ JavaScript, và sau đây xảy ra:

1. Lấy nội dung của tập tin "poll_result.txt"
2. Đưa nội dung của các tập tin trong các biến và thêm một biến được chọn
3. Viết kết quả các tập tin "poll_result.txt"
4. Sản xuất một đại diện đồ họa của các kết quả thăm dò ý kiến

Text File

Các tập tin văn bản (poll_result.txt) là nơi chúng tôi lưu trữ các dữ liệu từ các cuộc thăm dò.

Nó được lưu trữ như thế này:

```
0 | 0
```

Số đầu tiên đại diện cho "Yes" phiếu, số thứ hai đại diện cho phiếu "Không".

Lưu ý: nhớ để cho phép máy chủ web của bạn để chỉnh sửa các tập tin văn bản. Cung cấp cho truy cập tất cả mọi người, chỉ cần các máy chủ web (PHP).

Chapter Ngoài : Bài Tập

Bài 11: Viết ứng dụng đăng nhập bằng PHP và MYSQL



Ở những bài trước chúng ta đã học qua các kiến thức trọng yếu về **PHP**. Tuy nhiên, để có thể xây dựng một website hoàn chỉnh bằng những kiến thức đó thì quả thật không đơn giản. Bởi vì các kiến thức qua sách vở và tài liệu dù sao cũng chỉ là những kiến thức tổng quát và thiếu tính khách quan thực tiễn. Nên khi người học lập trình bước vào giai đoạn viết ứng dụng thì thường rất lúng túng. Đó cũng là lý do tôi viết bài này để hướng dẫn các bạn dần làm quen với cách tiếp cận một ứng dụng **PHP** và **MYSQL** như thế nào.

Trước hết chúng ta phải thiết kế và xây dựng mô hình cơ sở dữ liệu ứng với từng lệnh bên dưới (xem lại bài [**ngôn ngữ SQL và MYSQL Căn Bản**](#)) sau đó mới tiến hành triển khai lập trình.

```
1 mysql -hlocalhost -uroot -proot
2 mysql>create database project;
3 mysql>use project;
   mysql>create table user( id INT(10) UNSIGNED NOT NULL AUTO_INCREMENT,
4 username VARCHAR(50) NOT NULL, password CHAR(50) NOT NULL, level CHAR(1)
   NOT NULL, PRIMARY KEY(id));
5 mysql>insert into user(username,password,level) values
   ("admin","12345","2") ("abc","12345","1");
```

Vậy là chúng ta đã có cơ sở dữ liệu như mô hình trên. Tiếp đến chúng ta tạo file **login.php** và thiết kế Form HTML để có màn hình đăng nhập khi người dùng truy cập.

Username:

Password:

<http://www.qhonline.info>

```
1 <form action='login.php' method='post'>
2 Username: <input type='text' name='username' size='25' /><br />
3 Password: <input type='password' name='password' size='25' /><br />
4 <input type='submit' name='ok' value='Đăng Nhập' />
5 </form>
```

Tiếp đến chúng ta tiến hành kiểm tra dữ liệu từ form

```
01 <?php
02 if(isset($_POST['ok']))
03 {
04     $u=$p="";
05     if($_POST['username'] == NULL)
06     {
07         echo "Please enter your username<br />";
08     }
09     else
10     {
11         $u=$_POST['username'];
12     }
13     if($_POST['password'] == NULL)
14     {
15         echo "Please enter your password<br />";
16     }
17     else
18     {
19         $p=$_POST['password'];
20     }
21 }
22 ?>
```

Đoạn code ở trên sẽ kiểm tra xem người dùng có tiến hành nhấn nút đăng nhập hay không. Và nếu có thì chúng ta sẽ xét tiếp tình trạng người dùng có để trống các ô username và password hay không. Nếu có chúng ta sẽ thông báo lỗi ở bên trên form, để người sử dụng tiến hành nhập liệu. Vì phiên bản PHP 5.3 trở lên sẽ yêu cầu ta khai báo biến trước khi sử dụng. Vì thế mặc định ban đầu ta khởi tạo 2 biến \$u và \$p mang giá trị là rỗng.

Kế đến chúng ta kiểm tra xem có tồn tại hai biến \$u và \$p (chỉ khi người dùng đăng nhập thành công thì mới có thể tạo ra 2 biến đó). Tiếp đến chúng ta tiến hành kết nối cơ sở dữ liệu (xem lại bài [kết hợp PHP & MYSQL trong ứng dụng web](#)).

```
1 <?php
2 $conn=mysql_connect("localhost","root","root") or die("can't connect
  this database");
3 mysql_select_db("project",$conn);
4 ?>
```

Và tiến hành kiểm tra xem username và password người sử dụng vừa nhập có trùng khớp với thông tin có trong cơ sở dữ liệu hay không ?. Nếu không thì chúng ta sẽ báo lỗi ngay. Ngược lại sẽ tiến hành lấy dữ liệu từ bảng và gán vào **session**. Để có thể quản lý phiên làm việc một cách hiệu quả trên mọi trang của khu vực admin (xem lại bài [khái niệm cơ bản về cookie và session](#)).

```
01 <?php
02 if($u && $p)
03 {
04   $conn=mysql_connect("localhost","root","root") or die("can't connect
    this database");
05   mysql_select_db("project",$conn);
06   $sql="select * from user where username='".$u.'" and
    password='".$p.'";
07   $query=mysql_query($sql);
08   if(mysql_num_rows($query) == 0)
09   {
10     echo "Username or password is not correct, please try again";
11   }
12   else
13   {
14     $row=mysql_fetch_array($query);
15     session_start();
16     $_SESSION['userid'] = $row['id'];
17     $_SESSION['level'] = $row['level'];
18   }
19 }
20 ?>
```

Như vậy code hoàn chỉnh cho ứng dụng (file login.php) này là như sau:

```
01 <?php
02 if(isset($_POST['ok']))
03 {
```

```

04 $u=$p="";
05 if($_POST['username'] == NULL)
06 {
07     echo "Please enter your username<br />";
08 }
09 else
10 {
11     $u=$_POST['username'];
12 }
13 if($_POST['password'] == NULL)
14 {
15     echo "Please enter your password<br />";
16 }
17 else
18 {
19     $p=$_POST['password'];
20 }
21 if($u && $p)
22 {
23     $conn=mysql_connect("localhost","root","root") or die("can't
connect this database");
24     mysql_select_db("project",$conn);
25     $sql="select * from user where username='".$u."' and
password='".$p."'";
26     $query=mysql_query($sql);
27     if(mysql_num_rows($query) == 0)
28     {
29         echo "Username or password is not correct, please try again";
30     }
31     else
32     {
33         $row=mysql_fetch_array($query);
34         session_start();
35         $_SESSION['userid'] = $row[id];
36         $_SESSION['level'] = $row[level];
37     }
38 }
39 }
40 }
41 ?>
42 <form action='login.php' method='post'>
43 Username: <input type='text' name='username' size='25' /><br />
44 Password: <input type='password' name='password' size='25' /><br />

```

```
45 <input type='submit' name='ok' value='Dang Nhap' />
46 </form>
```

Ở bài sau, chúng ta sẽ tiếp tục tìm hiểu **ứng dụng kết hợp PHP & MYSQL cho việc thêm một thành viên** như thế nào. Mọi sự góp ý, cũng như thắc mắc của các bạn dành cho bài viết vui lòng sử dụng tính năng **thảo luận tại diễn đàn**. Hoặc thông qua việc **liên hệ của hệ thống website**. Mong sẽ nhận được nhiều ý kiến đóng góp từ các bạn.

Bài 12: Viết ứng dụng thêm thành viên bằng PHP và MYSQL



Ở bài trước, chúng ta đã xây dựng ứng dụng thành viên. Cho phép **người sử dụng thực thi công việc đăng nhập vào hệ thống**. Trong bài này chúng ta sẽ tiếp tục công việc phát triển ứng dụng trên, để người sử dụng sau khi đăng nhập có những thao tác đặc quyền như thêm, quản lý, sửa, xoá thành viên của trang web. (xem lại **Bài 11: Viết ứng dụng đăng nhập bằng PHP & MYSQL**).

Sau khi đăng nhập vào trang admin người dùng sẽ được đẩy sang trang quản lý admin, tại đây chỉ những ai đã đăng nhập đúng với username và password trong cơ sở dữ liệu và có quyền hạn level là 2 thì mới có thể truy xuất vào trang này. Vì vậy chúng ta cần thao tác **kiểm tra session để quản lý phiên làm việc của người sử dụng**. Đặt đoạn code sau ở những trang bạn muốn kiểm tra quyền truy cập của người sử dụng.

```
01 <?php
02 session_start();
03 if(isset($_SESSION['userid']) && $_SESSION['level'] == 2)
04 {
05 // tại đây thực thi các hoạt động khi đăng nhập thành công.
06 }
07 else
08 {
09 header("location: login.php");
10 exit();
11 }
12 ?>
```

Như vậy, ta tạo trang add_user.php và đặt đoạn code trên, đồng thời viết thêm 1 form thực thi công việc thêm thành viên.

Level:

Username:

Password:

Re-Password:

<http://www.qhonline.info>

```

01 <form action='add_user.php' method='POST'>
02 Level: <select name='level'>
03 <option value='1'>Member</option>
04 <option value='2'>Admin </option>
05 </select><br />
06 Username: <input type='text' name='username' size='25' /><br />
07 Password: <input type='password' name='password' size='25' /> <br />
08 Re-Password: <input type='password' name='re-password' size='25' /><br />
09 <input type='submit' name='adduser' value='Add New User' />
10 </form>

```

Tương tự như những bài trước, chúng ta cũng lần lượt kiểm tra từng đối tượng. Trước hết cần xem thử người dùng đã nhấn nút Add New user chưa ?, sau đó lại khai báo 2 biến \$u và \$p, mặc định ban đầu ta cho 2 biến này là rỗng. Kế tới ta kiểm tra xem người dùng đã nhập đầy đủ thông tin chưa. Cuối cùng là so sánh giữa password và re-password có giống nhau hay không. Tiếp theo chúng ta gán biến \$l sẽ bằng với giá trị mà khi người dùng chọn level cho user đó. (xem lại **Bài 4: Xử lý giá trị form trong PHP**)

```

01 <?php
02 if(isset($_POST['adduser']))
03 {
04     $u=$p="";
05     if($_POST['username'] == NULL)
06     {
07         echo "Vui long nhap username<br />";
08     }
09     else
10     {
11         $u=$_POST['username'];
12     }
13     if($_POST['password'] != $_POST['re-password'])
14     {
15         echo "Password va re-password khong chinh xac<br />";
16     }

```

```

17 else
18 {
19     if($_POST['password'] == NULL )
20     {
21         echo "Vui long nhap password<br />";
22     }
23     else
24     {
25         $p=$_POST['password'];
26     }
27 }
28 $l=$_POST['level'];
29 }
30 ?>

```

Kế đến chúng ta lại tiếp tục kiểm tra xem có tồn tại biến \$u, \$p và \$l hay không (chỉ khi người dùng nhập liệu thì mới tồn tại cùng lúc 3 biến này). sau khi đã kiểm tra sự tồn tại của 3 biến này chúng ta tiếp tục **dùng PHP kết nối cơ sở dữ liệu** để kiểm tra xem username mà người dùng vừa chọn có trùng với username khác trong cơ sở dữ liệu chưa. nếu trùng thì phải xuất thông báo lỗi yêu cầu họ chọn username khác còn nếu không thì ta lại tiến hành thêm thành viên mới này vào CSDL.

```

01 <?php
02 if($u & $p & $l)
03 {
04     $conn=mysql_connect("localhost","root","root") or die("can't connect
    this database");
05     mysql_select_db("project",$conn);
06     $sql="select * from user where username='".$u."'";
07     $query=mysql_query($sql);
08     if(mysql_num_rows($query) != "" )
09     {
10         echo "Username nay da ton tai roi<br />";
11     }
12     else
13     {
14         $sql2="insert into user(username,password,level)
        values('".$u."','".$p."','".$l."'");
15         $query2=mysql_query($sql2);
16         echo "Da them thanh vien moi thanh cong";
17     }
18 }
19 ?>

```

Tới đây ta đã kết thúc thao tác thêm một thành viên mới vào cơ sở dữ liệu. Sau đây là toàn bộ code trong ứng dụng này.

```
01 <?php
02 session_start();
03 if(isset($_SESSION['userid']) && $_SESSION['level'] == 2)
04 {
05     if(isset($_POST['adduser']))
06     {
07         $u = $p = "";
08         if($_POST['username'] == NULL)
09         {
10             echo "Vui long nhap username<br />";
11         }
12         else
13         {
14             $u=$_POST['username'];
15         }
16         if($_POST['password'] != $_POST['re-password'])
17         {
18             echo "Password va re-password khong chinh xac<br />";
19         }
20         else
21         {
22             if($_POST['password'] == NULL)
23             {
24                 echo "Vui long nhap password<br />";
25             }
26             else
27             {
28                 $p=$_POST['password'];
29             }
30         }
31         $l=$_POST['level'];
32         if($u & $p & $l)
33         {
34             $conn=mysql_connect("localhost","root","root") or die("can't
connect this database");
35             mysql_select_db("project",$conn);
36             $sql="select * from user where username='".$u."'";
37             $query=mysql_query($sql);
38             if(mysql_num_rows($query) != "" )
39             {
```

```

40     echo "Username nay da ton tai roi<br />";
41 }
42 else
43 {
44     $sql2="insert into user(username,password,level)
values('".$$u."', '".$$p."', '".$$l."')";
45     $query2=mysql_query($sql2);
46     echo "Da them thanh vien moi thanh cong";
47 }
48 }
49 }
50 }
51 ?>
52 <form action='add_user.php' method='POST'>
53 Level: <select name='level'>
54 <option value='1'>Member</option>
55 <option value='2'>Admin </option>
56 </select><br />
57 Username: <input type='text' name='username' size='25' /><br />
58 Password: <input type='password' name='password' size='25' /> <br />
59 Re-Password: <input type='password' name='re-password' size='25' /><br
/>
60 <input type='submit' name='adduser' value='Add New User' />
61 </form>
62 <?php
63 }
64 else
65 {
66     header("location: login.php");
67     exit();
68 }
69 ?>

```

Ở bài sau, chúng ta lại tiếp tục tìm hiểu ứng dụng kết hợp PHP và MYSQL cho việc **quản lý tất cả thành viên trong database** ra như thế nào. Mọi sự góp ý, cũng như thắc mắc của các bạn dành cho bài viết vui lòng sử dụng tính năng **thảo luận tại diễn đàn**. Hoặc thông qua việc **liên hệ của hệ thống website**. Mong sẽ nhận được nhiều ý kiến đóng góp từ các bạn.

Bài 13: Viết ứng dụng quản lý thành viên bằng PHP và MYSQL



Ở các bài trước, chúng ta đã cùng tìm hiểu về việc **thêm một thành viên** và **thiết lập trang đăng nhập vào hệ thống administrator**. Ở bài này, chúng ta sẽ cùng nghiên cứu về ứng

dụng quản trị các tài khoản. Hay nói cách khác là hệ thống lại toàn bộ danh sách của những thành viên trên website của chúng ta.

Sau khi đăng nhập vào trang admin người dùng sẽ được đẩy sang trang quản lý admin, tại đây chỉ những ai đã đăng nhập đúng với username và password trong cơ sở dữ liệu và có quyền hạn level là 2 thì mới có thể truy xuất vào trang này. Vì vậy chúng ta cần thao tác **kiểm tra session để quản lý phiên làm việc của người sử dụng**. Đặt đoạn code sau ở những trang bạn muốn kiểm tra quyền truy cập của người sử dụng.

```
01 <?php
02 session_start();
03 if(isset($_SESSION['userid']) && $_SESSION['level'] == 2)
04 {
05 // tại đây thực thi các hoạt động khi đăng nhập thành công.
06 }
07 else
08 {
09 header("location: login.php");
10 exit();
11 }
12 ?>
```

Như vậy, chúng ta sẽ đặt đoạn code này ở trang quản lý user. Cụ thể tôi đặt trang này có tên file là manage_user.php. Vì dữ liệu sẽ lặp lại toàn bộ user và ứng với từng user sẽ là 1 dòng dữ liệu được lặp lại. Chúng ta sẽ xây dựng 1 bảng gồm có STT là số thứ tự của từng user được đếm trên mỗi user khi lặp, username là tên truy cập của họ, level là cấp bậc của user (1 là member và 2 là admin), edit là cột chứa các link chỉnh sửa user, del là cột xóa các user.

```
1 <table align='center' width='400' border='1'>
2 <tr>
3 <td>STT</td>
4 <td>Username</td>
5 <td>Level</td>
6 <td>Edit</td>
7 <td>Del</td>
8 </tr>
```

Tiếp theo, chúng ta sẽ kết nối CSDL để tiến hành lập các user.

```
01 <?php
02 $conn=mysql_connect("localhost","root","root") or die("can't connect
```

```

        this database");
03 mysql_select_db("project",$conn);
04 $sql="select * from user order by id DESC";
05 $query=mysql_query($sql);
06 if(mysql_num_rows($query) == "")
07 {
08     echo "<tr><td colspan='5' align='center'>Chua co username
        nao</td></tr>";
09 }
10 ?>

```

Đoạn code ở trên tiến hành lựa chọn tất cả user có trong database. Đồng thời kiểm tra xem trong database có tồn tại user nào không. Nếu không sẽ xuất ra thông báo "chưa có username nào". Ngược lại khi có tồn tại user chúng ta sẽ xử lý tiếp như sau:

```

01 <?php
02 $stt=0;
03 while($row=mysql_fetch_array($query))
04 {
05     $stt++;
06     echo "<tr>";
07     echo "<td>$stt</td>";
08     echo "<td>$row['username']</td>";
09     if($row['level'] == "1")
10     {
11         echo "<td>Member</td>";
12     }
13     else
14     {
15         echo "<td>Admin</td>";
16     }
17     echo "<td><a href='edit_user.php?userid=$row[id]'>Edit</a></td>";
18     echo "<td><a href='del_user.php?userid=$row[id]'>Del</a></td>";
19     echo "</tr>";
20 ?>

```

Ở đây ta dùng biến \$stt để làm bộ đếm cho số lượng user. Như vậy, cứ mỗi lần lặp 1 user thì bộ đếm này sẽ tự động tăng lên 1 đơn vị. Sau đó, chúng ta đưa dữ liệu vào mảng và truyền các giá trị trên vào các cột dữ liệu trong bảng.

STT	Username	Level	Edit	Del
1	jacky	Admin	Edit	Del
2	test	Member	Edit	Del
3	abc	Member	Edit	Del
4	admin	Admin	Edit	Del

<http://www.qhonline.info>

Tới đây ta đã kết thúc thao tác quản lý các user đang tồn tại trong cơ sở dữ liệu. Sau đây là toàn bộ code trong ứng dụng này.

```

01 <?php
02 session_start();
03 if(isset($_SESSION['userid']) && $_SESSION['level'] == 2)
04 {
05 ?>
06 <table align='center' width='400' border='1'>
07 <tr>
08 <td>STT</td>
09 <td>Username</td>
10 <td>Level</td>
11 <td>Edit</td>
12 <td>Del</td>
13 </tr>
14 <?
15 $conn=mysql_connect("localhost","root","root") or die("can't connect
   this database");
16 mysql_select_db("project",$conn);
17 $sql="select * from user order by id DESC";
18 $query=mysql_query($sql);
19 if(mysql_num_rows($query) == "")
20 {
21 echo "<tr><td colspan=5 align=center>Chua co username nao</td></tr>";
22 }
23 else
24 {
25 $stt=0;
26 while($row=mysql_fetch_array($query))
27 {

```

```

28 $stt++;
29 echo "<tr>";
30 echo "<td>$stt</td>";
31 echo "<td>$row['username']</td>";
32 if($row['level'] == "1")
33 {
34 echo "<td>Member</td>";
35 }
36 else
37 {
38 echo "<td>Admin</td>";
39 }
40 echo "<td><a href='edit_user.php?userid=$row[id]'>Edit</a></td>";
41 echo "<td><a href='del_user.php?userid=$row[id]'>Del</a></td>";
42 echo "</tr>";
43 }
44
45 }
46 ?>
47 </table>

```

Ở bài sau, chúng ta lại tiếp tục tìm hiểu ứng dụng chỉnh sửa và xóa một thành viên trong PHP và MYSQL như thế nào. Mọi sự góp ý, cũng như thắc mắc của các bạn dành cho bài viết vui lòng sử dụng tính năng **thảo luận tại diễn đàn**. Hoặc thông qua việc **liên hệ của hệ thống website**. Mong sẽ nhận được nhiều ý kiến đóng góp từ các bạn.

Bài 14: viết ứng dụng sửa xóa thành viên bằng PHP và MYSQL



Trong bài trước chúng ta đã nói về cách thêm và quản lý thành viên kết hợp PHP và MYSQL. Tiếp theo trong bài này, chúng ta sẽ viết tiếp 2 ứng dụng sửa và xóa thành viên để hoàn tất module user.

Để theo kịp bài này, các bạn vui lòng xem lại bài 11,12,13. Vì các bài được hướng dẫn liên mạch từ đầu tới cuối.

Cũng như những bài trước, hệ thống sửa và xóa thành viên này cũng chỉ có thể thực hiện được khi người sử dụng đăng nhập thành công với quyền hạn của 1 administrator. Do vậy, bạn đừng quên kiểm tra session ở đầu khi bắt đầu viết ứng dụng này nhé.

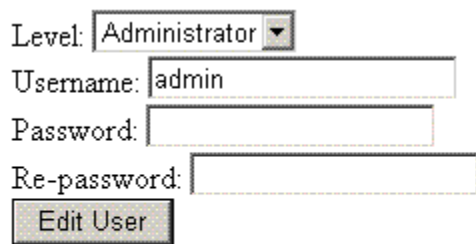
```

01 <?php
02 session_start();
03 if(isset($_SESSION['userid']) && $_SESSION['level'] == 2)
04 {
05     // tại đây thực thi các hoạt động khi đăng nhập thành công.
06 }
07 else
08 {
09     header("location: login.php");
10     exit();
11 }
12 ?>

```

A- Xây dựng trang sửa thành viên:

Vì là trang chỉnh sửa thành viên, nên nội dung của chúng có phần sẽ giống với phần thêm thành viên, chỉ khác là các ô nhập liệu giờ đây đã có dữ liệu. Dữ liệu này chúng ta tiến hành lấy từ cơ sở dữ liệu thông qua biến truyền mà ở trang quản lý đã gửi `edit_user.php?userid=$row[id]`



<http://www.qhonline.info>

Như vậy để lấy được giá trị từ liên kết này chúng ta sẽ sử dụng biến `$_GET['userid']`. Sau khi đã có được giá trị này, việc còn lại của bạn là lấy thông tin của id này từ cơ sở dữ liệu và đưa vào form để người dùng có thể chỉnh sửa.

Vậy ta có.

```

01 <?php
02 $sql="select * from user where id='".$_$id.'";
03 $query=mysql_query($sql);
04 $row=mysql_fetch_array($query);
05 ?>

```

```

06 <form action="edit_user.php?userid=<?=$id?>" method=post>
07 Level: <select name=level>
08   <option value=1 <? if($row[level] == 1) echo "selected"; ?>
    >Member</option>
09   <option value=2 <? if($row[level] == 2) echo "selected";
    ?>>Administrator</option>
10 </select><br />
11 Username: <input type=text name=user size=20
    value="<?=$row[username]?>" /><br />
12 Password: <input type=password name=pass size=20 /> <br />
13 Re-password: <input type=password name=repass size=20 /><br />
14 <input type=submit name=ok value="Edit User" />
15 </form>

```

Theo như đoạn code ở trên phần value chúng ta muốn đưa giá trị vào thì buộc phải sử dụng cú pháp gọn <?=\$ten?> hoặc cũng có thể sử dụng bằng cách <? echo \$ten; ?>.. Ở đây chúng ta chỉ đưa ra giá trị trên text box username, còn lại mật khẩu do có thể bị mã hóa nên không nên hiển thị ra. Phần level do lưu trong cơ sở dữ liệu là những con số 1,2 nên khi đưa ra chúng ta phải so sánh, nếu trong bảng lưu là 1 thì ở chỗ có giá trị bằng 1 sẽ thêm chữ selected. Để mặc định lựa chọn khi người dùng nhấn vào nút chỉnh sửa.

Tiếp tới người dùng sẽ nhấn nút edit. Việc còn lại của chúng ta là kiểm tra dữ liệu khi họ tiến hành chỉnh sửa dữ liệu. Điểm khác biệt ở đây là chúng ta phải cân nhắc khi viết trang chỉnh sửa. Bởi không phải ai vào chỉnh sửa cũng đều sửa cả thông tin username, password, level. Có khi họ chỉ sửa mỗi level hoặc có khi họ chỉ sửa mỗi password.

Vì thế ta phải xét các trường hợp để giải quyết.

Trường hợp 1: Người dùng chỉ chỉnh sửa username, level mà không chỉnh sửa password. Khi đó username, level bắt buộc đều đã có dữ liệu nên ta không cần kiểm tra chúng rỗng hay không. Nhưng với password chúng ta không thể bắt người lập trình nhập liệu. Và họ hoàn toàn có thể để rỗng.

Trường hợp 2: Người dùng chỉnh sửa username, level ,password. Khi đó ta xem password có dữ liệu và ta tiến hành cập nhật.

Để giải quyết bài toán trên thỏa 2 trường hợp ta sẽ xét phương pháp sau:

Ta kiểm tra password và repassword có giống nhau hay không ?. Nếu chúng khác nhau nghĩa là người dùng nhập liệu nhưng nhập sai. Vậy phải cảnh báo lỗi cho họ biết là họ đã nhập sai. Còn ngược lại nếu password và repassword giống nhau thì sẽ nảy sinh 2 trường hợp con.

+ Một là password và repassword đều có dữ liệu

+ Hai là password và repassword không có dữ liệu.

Khi đó ở trường hợp con 1 chúng ta sẽ khởi tạo biến \$p và ghi nhận thông tin họ nhập liệu. Và ở trường hợp 2 chúng ta không ghi nhận thông tin người nhập liệu vì họ không có ý định chỉnh sửa mật khẩu. Có thể viết đoạn code đơn giản như sau:

```
01 <?php
02 if($_POST['pass'] != $_POST['repass'])
03 {
04     echo "Password and re-password is not correct";
05 }
06 else
07 {
08     if($_POST['pass'] != NULL)
09     {
10         $p=$_POST['pass'];
11     }
12 }
13 ?>
```

Tiếp tới chúng ta sẽ phân loại dữ liệu cập nhật. Khi có username,password, level thì ta cập nhật tất cả thông tin của họ bằng cú pháp SQL. (xem lại [bài 9: ngôn ngữ SQL và MYSQL cơ bản](#)). Ngược lại khi không tồn tại password thì ta chỉ cập nhật username và level.

Một điều nữa chúng ta cần quan tâm đó là đối với lệnh update và delete, ta phải truyền 1 id cụ thể để tránh việc chúng xóa hoặc sửa dữ liệu toàn bộ trong bảng user. Với id tôi truyền nhận từ bên ngoài ta sẽ có.

```
01 <?php
02 if($u && $p && $l )
03 {
04     $sql="update user set username='".$u."', password='".$p."',
        level='".$l."' where id='".$id."'";
05     mysql_query($sql);
06     header("location:mana_user.php");
07     exit();
08 }
09 else
10 {
11     if($u && $l)
12     {
13         $sql="update user set username='".$u."', level='".$l."' where
            id='".$id."'";
14         mysql_query($sql);
15         header("location:mana_user.php");
16         exit();
17     }
18 }
19 ?>
```

Sau khi sửa thành công một thành viên chúng ta sẽ đưa người dùng trở về với trang quản lý (xem bài 13 để biết cách **xây dựng trang quản lý thành viên** như thế nào)

Nội dung của toàn bộ code ở trên như sau:

```
01 <?php
02 $conn=mysql_connect("localhost","root","root") or die("can't connect
    this database");
03 mysql_select_db("project",$conn);
04 $id=$_GET['userid'];
05 if(isset($_POST['ok']))
06 {
07     if($_POST['user'] == NULL)
08     {
09         echo "Please enter your username";
10     }
```

```

11 else
12 {
13     $u=$_POST['user'];
14 }
15 if($_POST['pass'] != $_POST['repass'])
16 {
17     echo "Password and re-password is not correct";
18 }
19 else
20 {
21     if($_POST['pass'] != NULL)
22     {
23         $p=$_POST['pass'];
24     }
25 }
26 $l = $_POST['level'];
27 if($u && $p && $l )
28 {
29     $sql="update user set username='".$u."', password='".$p."',
level='".$l."' where id='".$id."'";
30     mysql_query($sql);
31     header("location:mana_user.php");
32     exit();
33 }
34 else
35 {
36     if($u && $l)
37     {
38         $sql="update user set username='".$u."', level='".$l."' where
id='".$id."'";
39         mysql_query($sql);
40         header("location:mana_user.php");
41         exit();
42     }
43 }
44 }
45 $sql="select * from user where id='".$id."'";
46 $query=mysql_query($sql);
47 $row=mysql_fetch_array($query);
48 ?>
49 <form action="edit_user.php?userid=<?=$id?>" method=post>
50 Level: <select name=level>
51 <option value=1 <? if($row[level] == 1) echo "selected"; ?>)

```

```

    >Member</option>
52 <option value=2 <? if($row[level] == 2) echo "selected";
    ?>>Administrator</option>
53 </select><br />
54 Username: <input type=text name=user size=20
    value="<?=$row[username]?>" /><br />
55 Password: <input type=password name=pass size=20 /> <br />
56 Re-password: <input type=password name=repass size=20 /><br />
57 <input type=submit name=ok value="Edit User" />
58 </form>

```

B- Xây dựng trang xóa thành viên:

Đối với trang xóa dữ liệu, chúng ta cũng không cần phải xử lý quá nhiều. Bởi nhiệm vụ của chúng chỉ đơn giản là xóa đi những dòng trong bảng.

Như vậy cũng như trang edit chúng ta nhận giá trị từ nội dung liên kết ở trang quản lý đã gửi là `del_user.php?userid=$row[id]`.

Và thực thi lệnh kết nối csdl để xử lý nội dung này.

```

1 <?php
2 $conn=mysql_connect("localhost","root","root") or die("can't connect
  this database");
3 mysql_select_db("project",$conn);
4 $sql="delete from user where id='".$id."'";
5 mysql_query($sql);
6 header("location:mana_user.php");
7 exit();
8 ?>

```

Sau khi xóa xong thành viên thì hệ thống sẽ đưa chúng ta trở về trang quản lý user lại.

C- Tổng kết:

Như vậy chúng ta đã kết thúc các thao tác thêm, sửa, xóa, quản lý thành viên. Qua những chuỗi bài học này, chắc hẳn các bạn đã có những khái niệm hình thành từ việc ứng dụng lập trình PHP để xây dựng các hệ thống cơ bản rồi. Việc lập trình 1 ứng dụng đòi hỏi bạn ngoài việc vận dụng nội dung đã học ra, bạn còn phải biết linh động xử lý các tình huống một cách hiệu quả. Tránh việc rập khuôn một cách máy móc và học thuộc các nội dung code. Vì ở những mô hình khác, bài toán khác bạn sẽ có những cách thức xử lý và giải quyết khác nhau.

Mọi sự góp ý, cũng như thắc mắc của các bạn dành cho bài viết vui lòng sử dụng tính năng **thảo luận tại diễn đàn**. Hoặc thông qua việc **liên hệ của hệ thống website**. Mong sẽ nhận được nhiều ý kiến đóng góp từ các bạn.

Bài 15: Viết ứng dụng tạo mã xác nhận bằng PHP



Ở những bài trước, chúng ta đã tìm hiểu về cách làm việc với PHP kết hợp MYSQL trong việc xây dựng hệ thống thêm, sửa, xóa và quản lý user. Trong bài này, chúng ta tiếp tục tìm hiểu phương pháp tạo dãy số ngẫu nhiên để chống tấn công flood dữ liệu. Hoặc ai đó cố tình spam khiến cơ sở dữ liệu của chúng ta không thể xử lý tiếp được.

Trước tiên, chúng ta tiến hành tạo form HTML đơn giản để thực thi thao tác nhập liệu như sau:

```
01 <form action="form.php" method=post>
02 <table>
03   <tr>
04     <td align="left">
05       <label for="captcha">Captcha</label>
06     </td>
07     <td>
08       <input type="text" name="txtCaptcha" maxlength="10" size="32" />
09     </td>
10     <td>
11       
12     </td>
13   </tr>
14 </table>
15   <td> </td>
16   <td>
```

```

17     <input type=submit name=ok value="Check" />
18   </td>
19 </tr>
20 </table>
21 </form>

```

Chúng ta chú ý phần hình ảnh, tại đây tôi truyền đường dẫn chính là liên kết tới trang `random_image.php`. Trang này sẽ thực thi công việc tạo ra những bức hình có dãy số ngẫu nhiên để phân nhập liệu của người sử dụng tham chiếu tới.

Tiếp theo, chúng ta khởi tạo file `random_image.php` để lấy ra dãy số ngẫu nhiên và phát sinh chúng ngay trên tấm hình cho người truy cập nhập liệu.

Để làm được điều đó. Chúng ta sẽ khởi tạo session và lưu dãy số nhập liệu đó vào session của mình, sau đó so khớp với trang form bên kia.(xem lại [bài 7: Khái niệm cơ bản về cookie và session trong PHP](#))



Đầu tiên chúng ta sử dụng hàm `md5` và `rand` để mã hóa các ký tự bao gồm số và chữ cái. Khi sử dụng hàm `md5()` ký tự phát sinh sẽ lên tới 32 ký tự. Và chúng ta chỉ lấy duy nhất 5 ký tự từ chuỗi mã hóa đó bằng hàm `substr`. Tiếp tục ta lưu đoạn mã hóa này trong session cụ thể `$_SESSION['security_code']`, để tại trang form ta sẽ sử dụng so sánh với phần nhập liệu của người sử dụng.

Vậy nên đoạn code dưới sẽ giải quyết các tình huống này.

```

1 <?php
2 $md5_hash = md5(rand(0,999));
3 $security_code = substr($md5_hash, 15, 5);
4 $_SESSION["security_code"] = $security_code;
5 ?>

```

Kế đến ta khởi tạo tấm hình với chiều rộng và chiều cao mà ta thiết lập thông qua hàm `ImageCreate()` . Và tiếp tục khai báo 2 màu chính là trắng và đen bằng hàm `ImageColorAllocate($image, red, green, blue)`. Hàm này sẽ tạo ra một màu sắc từ hệ màu RGB trên tấm hình mà chúng ta vừa khởi tạo.

Tiếp tục ta đổ background của tấm hình sẽ là màu đen và chữ xuất hiện trên tấm hình sẽ là màu trắng bằng hàm `ImageFill()` và hàm `ImageString()`. Hàm `ImageString` có một số đối số cơ bản như sau: `ImageString($image, 5, 30, 6, $security_code, $white)`; Trong đó:

+ `$image` là hình mà chúng ta khởi tạo.

+ `5`: là font-size mà chúng ta quy ước cho ký tự xuất hiện trên hình.

+ `30`: là khoảng cách bên trái của tấm hình.

+ `6` : là khoảng cách từ trên của tấm hình.

+ `$security_code`: là đoạn code sau khi chúng ta cắt ra ở trên.

+ `$white`: là màu sắc mà chúng ta đã sử dụng hàm `ImageColorAllocate()` ở trên để khởi tạo ra màu trắng.

```
1 <?php
2 $width = 100;
3 $height = 30;
4 $image = ImageCreate($width, $height);
5 $white = ImageColorAllocate($image, 255, 255, 255);
6 $black = ImageColorAllocate($image, 0, 0, 0);
7 ImageFill($image, 0, 0, $black);
8 ImageString($image, 5, 30, 6, $security_code, $white);
9 ?>
```

Kể tới chúng ta sẽ sử dụng lệnh header để trả nội dung này trở về dữ liệu dạng hình. Và tiến hành khởi tạo định dạng cho file hình mà chúng ta vừa tạo là JPG đồng thời giải phóng hình tạm được sử dụng để khởi tạo ra tấm hình này bằng đoạn code sau:

```
1 <?php
2 header("Content-Type: image/jpeg");
3 ImageJpeg($image);
4 ImageDestroy($image);
5 ?>
```

Để dễ quản lý chúng ta sẽ đặt tất cả đoạn code trên vào trong 1 hàm để dễ sử dụng và quản lý chúng tốt hơn. Vậy toàn bộ đoạn code trong trang random_image.php này như sau.

```
01 <?php
02 session_start();
03 function create_image()
04 {
05     $md5_hash = md5(rand(0,999));
06     $security_code = substr($md5_hash, 15, 5);
07     $_SESSION["security_code"] = $security_code;
08     $width = 100;
09     $height = 30;
10     $image = ImageCreate($width, $height);
11     $white = ImageColorAllocate($image, 255, 255, 255);
12     $black = ImageColorAllocate($image, 0, 0, 0);
13     ImageFill($image, 0, 0, $black);
14     ImageString($image, 5, 30, 6, $security_code, $white);
15     header("Content-Type: image/jpeg");
16     ImageJpeg($image);
17     ImageDestroy($image);
18 }
19 create_image();
20 exit();
21 ?>
```

Tiếp tục, trong trang form chúng ta phải khởi tạo session để chúng nhận được các giá trị mà ta đã khởi tạo và sử dụng ở trang random_image, kể tới ta kiểm tra xem người dùng có nhấn submit chưa. Nếu có ta sẽ tiếp tục kiểm tra xem người dùng nhập vào textbox có đúng là dãy số hiển thị trên hình ảnh hay không. Và xuất ra thông báo tương ứng với nội dung của người nhập liệu. (xem lại [bài 4: Xử lý giá trị form trong PHP](#))

Sau đây là toàn bộ code của trang form.php.

```
01 <?php
02 session_start();
03 if(isset($_POST['ok']))
04 {
05     if($_POST['txtCaptcha'] == NULL)
06     {
07         echo "Please enter your code";
08     }
09     else
10     {
11         if($_POST['txtCaptcha'] == $_SESSION['security_code'])
12         {
13             echo "ma lenh hop le";
14         }
15         else
16         {
17             echo "Ma lenh khong hop le";
18         }
19     }
20 }
21 ?>
22 <form action="form.php" method=post>
23 <table>
24 <tr>
25     <td align="left">
26         <label for="captcha">Captcha</label>
27     </td>
28     <td>
29         <input type="text" name="txtCaptcha" maxlength="10" size="32" />
30     </td>
31     <td>
32         
33     </td>
34 </tr>
35 <tr>
36     <td> </td>
37     <td>
38         <input type=submit name=ok value="Check" />
39     </td>
40 </tr>
41 </table>
```

Download toàn bộ source code này [tại đây](#).

Tổng kết:

Qua bài học này, chúng ta đã hiểu được nguyên tắc xử lý và những hàm cơ bản mà 1 hệ thống khi sử dụng captcha cần phải vận dụng như thế nào. Và tất nhiên hàm xử lý captcha này cũng chỉ mang kiến trúc tương đối. Để tăng cường tính bảo mật hơn, các bạn nên sử dụng 1 số hàm chuyên dụng nhằm mã hóa hoặc gây nội dung khó nhìn hơn để chặn các tools hoặc các hệ thống bot có thể qua mặt thông số này.

Mọi sự góp ý, cũng như thắc mắc của các bạn dành cho bài viết vui lòng sử dụng tính năng [thảo luận tại diễn đàn](#). Hoặc thông qua việc [liên hệ của hệ thống website](#). Mong sẽ nhận được nhiều ý kiến đóng góp từ các bạn.

Bài 16: Viết ứng dụng đếm số người online bằng php



Trong bài này, chúng ta sẽ nói về cách thức xây dựng tính năng thống kê số người đang viếng thăm website của bạn. Vì là chức năng nhỏ, nên tôi không đi nhiều về những tính năng lớn như bao nhiêu khách, bao nhiêu thành viên,...Chỉ đơn thuần là quá trình hiển thị hiện tại có bao nhiêu người đang thăm viếng website của bạn.

Trước hết, chúng ta khởi tạo cơ sở dữ liệu như sau. (xem lại bài [bài 9: ngôn ngữ SQL và MYSQL cơ bản](#))

```
1 mysql> create table useronline (tgtmp INT(15) DEFAULT "0" NOT NULL ,
2 ip VARCHAR(50) NOT NULL ,
3 local VARCHAR(100) NOT NULL,
4 PRIMARY KEY (tgtmp) ,
5 KEY ip(ip) ,
6 KEY local(local));
```

tgtmp là thời gian mà họ truy cập được tính khi họ truy vào trang web đó.

IP là dãy số lưu thông tin IP của họ khi viếng thăm website của chúng ta.

local là nơi lưu đường dẫn mà họ đang truy cập.

Khi người dùng truy cập vào trang web, chúng ta sẽ tiến hành insert thông tin của họ vào cơ sở dữ liệu với các tham số cơ bản nhưng trong database.

tgtmp được tính ra bằng hàm time(). Hàm này sẽ lấy ra thời gian hiện tại của người truy cập. Tiếp tục ta lại tính thời gian mới của họ, được tính bằng việc quy ước thời gian quy định. Cụ thể ở đây tôi cho là 900 giây tương đương với 15 phút truy cập của họ.

```
1 <?php
2 $tg=time();
3 $tgout=900;
4 $tgnew=$tg - $tgout;
5 ?>
```

như vậy nếu thời gian lưu trong database mà nhỏ hơn thời gian new này thì chúng ta có thể hiểu rằng vị khách ấy đã rời khỏi website của chúng ta. Cụ thể hơn.

Ví dụ: tôi viếng thăm website đó là 7h.

như vậy hệ thống sẽ ghi thông tin lúc đó là 7h.

Nếu sau 1 thời gian tôi không làm gì, hoặc tôi không truy cập website đó nữa thì hệ thống sẽ không ghi nhận thông tin mới. Như thế nếu bây giờ 8h và trừ đi 15 phút tôi quy ước, rõ ràng là thời gian mới hiện tại là 7h45, Thời gian này vẫn lớn hơn thời điểm lưu thông tin 7h ($7h45 > 7h$). Do vậy, nếu chúng thỏa điều kiện đó thì chúng ta chỉ việc xóa đi các record trong cơ sở dữ liệu là xong.

Vậy ta có code kết nối CSDL như sau:(xem lại **bài 10: kết hợp PHP và MYSQL trong ứng dụng**)

```
1 <?php
2 $conn=mysql_connect("localhost","root","root") or die("can't
  connect");
3 mysql_select_db("online",$conn);
4 ?>
```

Sau đó ta tiến hành ghi nhận thông tin người dùng vào CSDL.

```
1 <?php
2 $sql="insert into useronline(tgtmp,ip,local)
  values('$tg','$REMOTE_ADDR','$PHP_SELF')";
3 $query=mysql_query($sql);
4 ?>
```

\$REMOTE_ADDR là biến môi trường dùng để lấy ra IP của người truy cập.

\$PHP_SELF là biến môi trường dùng để lấy ra đường dẫn mà người dùng đang truy cập.

Tiếp đến ta tiến hành xóa record khi thời gian thực lớn hơn thời gian trong cơ sở dữ liệu.

```
1 <?php
2 $sql="delete from useronline where tgtmp < $tgnew";
3 $query=mysql_query($sql);
4 ?>
```

Tiếp tục là công việc hiển thị thông tin ra bên ngoài bằng cách liệt kê các record có trong database.

```
1 <?php
2 $sql="SELECT DISTINCT ip FROM useronline WHERE local='$PHP_SELF'";
3 $query=mysql_query($sql);
4 $user = mysql_num_rows($query);
5 ?>
```

DISTINCT là cú pháp cho phép liệt kê các dòng record mà không cho phép chúng có dữ liệu trùng lặp như cú pháp select bình thường.

Vậy chúng ta sẽ liệt kê tất cả những ip của những ai đang truy cập trên trang useronline.php.

Phần việc cuối cùng còn lại là chúng ta sẽ xuất thông tin ấy ra trình duyệt.

Và sau đây là toàn bộ nội dung code của trang useronline.php

```
01 <?php
02 $tg=time();
03 $tgout=900;
04 $tgnew=$tg - $tgout;
05 $conn=mysql_connect("localhost","root","root") or die("can't
   connect");
06 mysql_select_db("online",$conn);
07 $sql="insert into useronline(tgtmp,ip,local)
   values('$tg','$REMOTE_ADDR','$PHP_SELF')";
08 $query=mysql_query($sql);
09 $sql="delete from useronline where tgtmp < $tgnew";
10 $query=mysql_query($sql);
11 $sql="SELECT DISTINCT ip FROM useronline WHERE local='$PHP_SELF'";
12 $query=mysql_query($sql);
13 $user = mysql_num_rows($query);
14 echo "user online :$user";
15 ?>
```

Bài 17: Xây dựng hệ thống bình chọn bằng PHP và MYSQL



Ở loạt bài trước, chúng ta đã bàn tới các ứng dụng quản lý người sử dụng, xây dựng bộ đếm, tạo mã bảo mật để phòng chống những người có dụng ý xấu trên website. Tiếp theo, trong bài này chúng ta sẽ bàn về cách thức xây dựng hệ thống cho phép người dùng bình chọn khi họ tiến hành thăm viếng website của chúng ta.

Để xây dựng hệ thống bình chọn chúng ta cần phân tích xem hệ thống này cần có bao nhiêu bảng quan hệ.

A- Phân tích và thiết kế cơ sở dữ liệu:

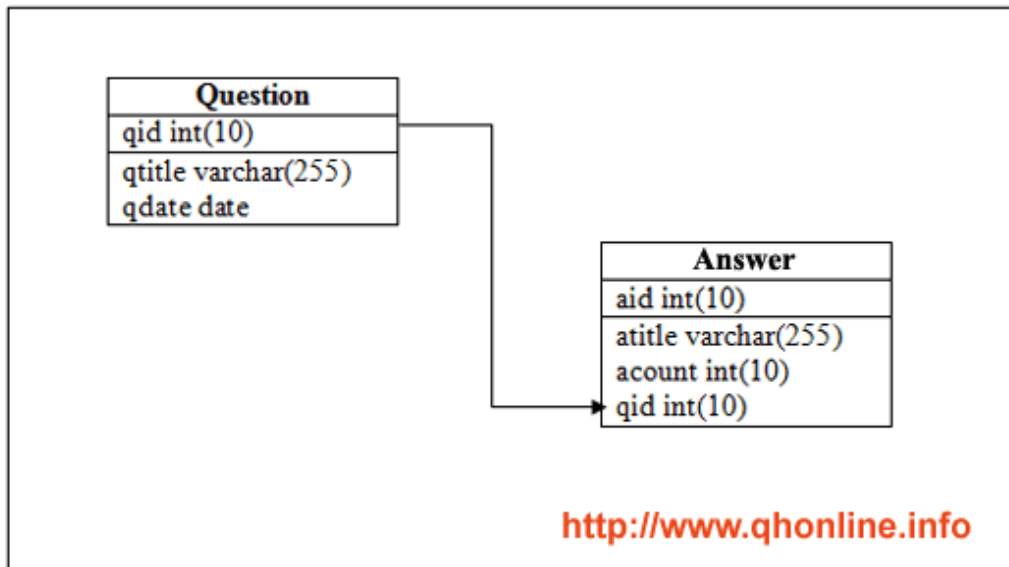
Trong tình huống này chúng ta cần 1 bảng câu hỏi và 1 bảng câu trả lời.

Phân tích quan hệ giữa chúng ta có:

1 Câu hỏi có nhiều câu trả lời.

1 câu trả lời chỉ dành cho 1 câu hỏi.

Vậy giữa 2 bảng này sẽ phát sinh một khóa ngoại, là khóa dùng để liên kết giữa hai bảng này.



Vậy ta có cú pháp tạo 2 bảng như sau.

Bảng câu hỏi (question)

```
mysql> create table question(qid INT(10) UNSIGNED NOT NULL
1 AUTO_INCREMENT,
2 -> qtitle VARCHAR(255) NOT NULL,
3 -> qdate DATE NOT NULL DEFAULT '0000-00-00',
```

```
4 -> PRIMARY KEY(qid));
```

Bảng câu trả lời (answer)

```
1 mysql> create table answer(aid INT(10) UNSIGNED NOT NULL AUTO_INCREMENT,  
2 -> qid INT(10) UNSIGNED NOT NULL,  
3 -> atitle VARCHAR(255) NOT NULL,  
4 -> account INT(10) NOT NULL DEFAULT '0',  
5 -> PRIMARY KEY(aid));
```

Nhập liệu bằng CSDL ta có:

```
1 mysql> Insert into question(qtitle,qdate) values("Ban thay qhonline the  
   nao ?", "  
2 2009-10-10");  
3 mysql> Insert into answer(qid,atitle,account) values("1","Nhin rat  
   dep",0);  
4 mysql> Insert into answer(qid,atitle,account) values("1","Nhin Dep",0);  
5 mysql> Insert into answer(qid,atitle,account) values("1","Nhin Cung  
   duoc",0);  
6 mysql> Insert into answer(qid,atitle,account) values("1","Nhin qua  
   xau",0);
```

B- Xây dựng trang bình chọn poll.php

Kết nối cơ sở dữ liệu:

```
1 <?php  
2 $conn=mysql_connect("localhost","root","root") or die("can not connect  
   database");  
3 mysql_select_db("poll_exam",$conn);  
4 ?>
```

Lựa chọn câu hỏi có trong cơ sở dữ liệu để liệt kê ra trên website, sau đó ta lại tiếp tục liệt kê các câu trả lời của câu hỏi đó bằng cách sử dụng dấu chọn lựa (radio). Như vậy chúng ta sẽ phải chạy cùng lúc 2 câu truy vấn lồng nhau trong suốt quá trình truy xuất.

```
01 <?php  
02 $sql="select * from question order by qid desc";  
03 $query=mysql_query($sql);  
04 if(mysql_num_rows($query) > 0)  
05 {  
06   $row=mysql_fetch_array($query);  
07   $qid=$row[qid];
```

```

08 echo "<form action='poll.php?questionid=$qid' method='post' >";
09 echo "<h2>$row[qttitle]</h2>";
10 $sql2="select * from answer where qid='$qid' order by aid";
11 $query2=mysql_query($sql2);
12 if(mysql_num_rows($query2) > 0)
13 {
14     while($row2=mysql_fetch_array($query2)){
15         echo "<input type='radio' name='answer'
16         value=$row2[aid]>$row2[atitle]<br />";
17     }
18     echo "<input type='submit' name='ok' value='Bình Chọn'>";
19     echo "<a href='result.php?questionid=$qid'>Xem Ket Qua</a>";
20     echo "</form>";
21 }
22 ?>

```

Và màn hình sẽ xuất ra như hình bên dưới

Ban thay qhonline the nao ?

☐ Nhìn rat dep
☐ Nhìn Dep
☐ Nhìn Cung duoc
☐ Nhìn qua xau
 [Xem Ket Qua](#)

<http://www.qhonline.info>

Và khi người dùng nhất nút Bình chọn, chúng ta sẽ gọi lại chính trang đó để xử lý dữ liệu mà người dùng vừa lựa chọn.

Vậy ta phải sử dụng lệnh isset() để kiểm tra xem người dùng có nhấn nút bình chọn không, tiếp đến ta lấy ra id mà người dùng vừa tiến hành chọn ở form bên dưới.

Cuối cùng ta cập nhật dữ liệu bằng cách lấy số trong cơ sở dữ liệu cộng tiếp cho 1 đơn vị nữa. Và đưa người dùng sang trang kết quả.

Vì trong đoạn code này có lệnh header nên chúng ta phải đặt quá trình xử lý này ở trên form. Nếu không sẽ bị lỗi dữ liệu khi gọi.

```
01 <?php
02 if (isset($_POST['ok']))
03 {
04     $id=$_POST['answer'];
05     $qid=$_GET['questionid'];
06     $sql3="update answer set account=account + 1 where aid='".$id."'";
07     mysql_query($sql3);
08     header("location: result.php?questionid=$qid");
09     exit();
10 }
11 ?>
```

Vậy toàn bộ code của trang poll.php này như sau.

```
01 <?php
02 $conn=mysql_connect("localhost","root","root") or die("can not
03 connect database");
04 mysql_select_db("poll_exam",$conn);
05 if (isset($_POST['ok']))
06 {
07     $id=$_POST['answer'];
08     $qid=$_GET['questionid'];
09     $sql3="update answer set account=account + 1 where aid='".$id."'";
10     mysql_query($sql3);
11     header("location: result.php?questionid=$qid");
12     exit();
13 }
14 $sql="select * from question order by qid desc";
15 $query=mysql_query($sql);
16 if(mysql_num_rows($query) > 0)
17 {
18     $row=mysql_fetch_array($query);
19     $qid=$row[qid];
20     echo "<form action='poll.php?questionid=$qid' method='post' >";
21     echo "<h2>$row[qttitle]</h2>";
22     $sql2="select * from answer where qid='$qid' order by aid";
```

```

22 $query2=mysql_query($sql2);
23 if(mysql_num_rows($query2) > 0)
24 {
25     while($row2=mysql_fetch_array($query2)){
26         echo "<input type=radio name=answer
value=$row2[aid]>$row2[atitle]<br />";
27     }
28 }
29 echo "<input type='submit' name='ok' value='Bình Chọn'>";
30 echo "<a href='result.php?questionid=$qid'>Xem Ket Qua</a>";
31 echo "</form>";
32 }
33 ?>

```

C- Xây dựng trang kết quả bình chọn result.php

Ở trang này, ta sẽ lấy giá trị question id của người bình chọn từ trang poll.php và tính toán xem tỷ lệ phần trăm của các câu hỏi ấy như thế nào.

Đầu tiên, ta lấy giá trị từ poll.php?questionid=1. vậy 1 là giá trị ta sẽ có được từ \$_GET['questionid']. Tiếp đến ta lại dùng giá trị vừa lấy được này để liệt kê thông tin câu hỏi và tính toán giá trị.

Để tính toán giá trị, ta cần biết tổng số bình chọn trong toàn bộ các câu trả lời hiện nay là bao nhiêu. Bằng câu lệnh SUM ta có thể làm được điều đó.

```

1 $sql2="select qid, SUM(acount) as total from answer group by qid
having qid='".$qid."'";

```

Câu lệnh này cho phép ta lấy ra tổng số bình chọn trong cơ sở dữ liệu của câu hỏi mà ta truyền vào. Để SUM được account ta cần phải GROUP(gom nhóm) chúng ta lại theo mã câu hỏi mà chúng ta cần tìm. having là mệnh đề theo sau có ý nghĩa tương đương với where, nó thường được sử dụng theo GROUP.

Như vậy để lấy ra tổng số phiếu bình chọn ta chỉ việc thực thi câu truy vấn này, và lấy tên cột tạm là total.

```

1 <?php
2 $sql2="select qid, SUM(acount) as total from answer group by qid
   having qid='".$qid."'";
3 $query2=mysql_query($sql2);
4 $row2=mysql_fetch_array($query2);
5 $total=$row2[total];
6 ?>

```

kế tới, ta lại tính toán số phiếu của từng câu hỏi trong cơ sở dữ liệu, dựa vào tổng số câu hỏi này.

Ví dụ: Tổng số lượt bình chọn là 10

câu 1 là 3

câu 2 là 4

câu 3 là 2

câu 4 là 1

Vậy suy ra phần trăm trên tổng số 10 của 4 câu này được tính bằng cách lấy số liệu của từng câu chia cho tổng số câu hỏi và nhân cho 100. Ví dụ câu 1 là: $(3/10)*100 = 30\%$

Để làm tròn kết quả ta sử dụng hàm round(). Giúp dữ liệu đưa về sự đồng bộ và rõ ràng nhất.

Vậy code xử lý của chúng ta như sau:

```

01 <?php
02 $sql3="select * from answer where qid='".$qid."' order by aid";
03 $query3=mysql_query($sql3);
04 if(mysql_num_rows($query3) > 0)
05 {
06   while($row3=mysql_fetch_array($query3)) {
07     $percent=round(($row3[acount]/$total)*100,2);
08     echo "<h4 style='color:red; font:12px verdana; '>$row3[atitle] :
   $row3[acount] ($percent %) </h4>";
09   }

```

```
10 }
11 ?>
```

Kết quả khi xuất ra sẽ có hình như sau:

Ban thay qhonline the nao ?

Nhin rat dep : 1 (14.29 %)

Nhin Dep : 4 (57.14 %)

Nhin Cung duoc : 1 (14.29 %)

Nhin qua xau : 1 (14.29 %)

<http://www.qhonline.info>

Vậy toàn bộ mã nguồn của trang result.php này như sau:

```
01 <?php
02 $conn=mysql_connect("localhost","root","root") or die("can not
   connect database");
03 mysql_select_db("poll_exam",$conn);
04 if(isset($_GET['questionid']))
05 {
06   $qid=$_GET['questionid'];
07   $sql="select * from question where qid='".$qid."'";
08   $query=mysql_query($sql);
09   $row=mysql_fetch_array($query);
10   echo "<h2>$row[qttitle]</h2>";
11   $sql2="select qid, SUM(acount) as total from answer group by qid
   having qid='".$qid."'";
12   $query2=mysql_query($sql2);
13   $row2=mysql_fetch_array($query2);
14   $total=$row2[total];
15   $sql3="select * from answer where qid='".$qid."' order by aid";
16   $query3=mysql_query($sql3);
17   if(mysql_num_rows($query3) > 0){
18     while($row3=mysql_fetch_array($query3)){
19       $percent=round(($row3[acount]/$total)*100,2);
20       echo "<h4 style='color:red; font:12px verdana; '>$row3[atitle] :
   $row3[acount] ($percent %) </h4>";
```

```

21 }
22 }
23 }
24 ?>

```

Download toàn bộ mã nguồn của bài này [tại đây](#).

Mọi sự góp ý, cũng như thắc mắc của các bạn dành cho bài viết vui lòng sử dụng tính năng [thảo luận tại diễn đàn](#). Hoặc thông qua việc [liên hệ của hệ thống website](#). Mong sẽ nhận được nhiều ý kiến đóng góp từ các bạn.



Bài 18: Xây dựng ứng dụng tạo trang upload nhiều hình ảnh

Ở những bài trước, chúng ta đã có dịp nghiên cứu từng ứng dụng nhỏ bởi việc kết hợp PHP và Mysql. Trong bài này, chúng ta sẽ tiếp tục tìm hiểu phương pháp xây dựng một trang upload nhiều hình ảnh như thế nào. Và lưu chúng vào cơ sở dữ liệu ra sao trong lập trình php.

Mô hình này là rất cần thiết đối với các hệ thống. Bởi qua đó người sử dụng có thể thao tác và upload những hình ảnh đi kèm với hệ thống của họ như tin tức, sản phẩm,...

Đầu tiên, ta tiến hành xây dựng cơ sở dữ liệu với bảng images như sau (xem lại [bài 9: cơ bản về ngôn ngữ sql và mysql](#)).

```

1 CREATE TABLE `images` (
2   `img_id` INT( 10 ) UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY ,
3   `img_url` VARCHAR( 255 ) NOT NULL ,
4   `img_name` VARCHAR( 150 ) NOT
5   NULL
6 );

```

Như vậy, tôi có bảng images với 3 cột thuộc tính là img_id dùng để lưu khóa của từng hình, img_url dùng để lưu đường dẫn vào hệ thống, img_name dùng để lưu tên hình ảnh.

Ứng dụng của chúng ta cũng rất cần sự tùy biến và dễ sử dụng. Sao cho người dùng có thể hoàn toàn thao tác. Ví dụ: nếu có 2 hình, họ chỉ muốn tạo ra 2 chế độ chọn file thì sẽ chỉ khai báo là 2. Và ngược lại nếu là n hình, họ sẽ tạo được n file lựa chọn. Để làm được điều này, trước tiên ta cần tạo 1 form nhỏ cho phép người dùng nhập vào số lượng file mà họ muốn thao tác. Vậy tôi khởi tạo file có tên upload.php với mã lệnh form như sau:

```

1 <form action="upload.php" method="post">
2   Enter your Image: <input type="text" name="txtnum" value="<?php echo
3   $_POST['txtnum']; ?>" size="10" />

```

```

3 <input type="submit" name="ok_num" value="Accept" />
4 </form>

```

Sau khi người dùng nhấn nút accept thì cũng là lúc chúng ta tạo ra số field upload theo yêu cầu của họ. Vậy, ta có code xử lý bên dưới như sau: (xem lại [bài 4: xử lý giá trị trên form](#)).

```

01 <?php
02 if(isset($_POST['ok_num']))
03 {
04     $num=$_POST['txtnum'];
05     echo "<hr />";
06     echo "Ban dang chon $num file upload<br />";
07     echo "<form action='doupload.php?file=$num' method='post'
08 enctype='multipart/form-data'>";
09     for($i=1; $i <= $num; $i++)
10     {
11         echo "<input type='file' name='img[]' /><br />";
12     }
13     echo "<input type='submit' name='ok_upload'
14 value='Upload' />";
15     echo "</form>";
16 }
17 ?>

```

Đoạn code này sử dụng vòng lặp for với điều kiện là biến \$i sẽ nhỏ hơn số file mà họ nhập vào. Đồng thời, để việc upload thành công tại form ta cũng cần thêm đoạn code :

`enctype='multipart/form-data'.`

Số lượng field sẽ lặp theo yêu cầu của người dùng. Do vậy, tại name của thẻ input, ta sẽ sử dụng mảng để lưu thông tin. Sử dụng name='img[]', có nghĩa là mỗi lần lặp và mỗi lần người dùng chọn file thì mảng img sẽ thêm 1 khóa tương ứng. (xem lại [bài 6: tổng quan về mảng và các hàm hỗ trợ](#))

Và cuối cùng là nút submit thực thi công việc upload và gọi tới trang doupload.php?file=\$num (chỉ rõ số file cụ thể do người dùng nhập tại trang doupload).

Enter your Image: www.qhonline.info

Ban dang chon 5 file upload

<input type="button" value="Choose File"/>	no file selected
<input type="button" value="Choose File"/>	no file selected
<input type="button" value="Choose File"/>	no file selected
<input type="button" value="Choose File"/>	no file selected
<input type="button" value="Choose File"/>	no file selected
<input type="button" value="Upload"/>	

www.qhonline.info

Vậy toàn bộ code của trang upload này sẽ như sau:

```
01 <form action="upload.php" method="post">
02 Enter your Image: <input type="text" name="txtnum" value="<?php echo
03 $_POST['txtnum']; ?>" size="10" />
04 <input type="submit" name="ok_num" value="Accept" />
05 </form>
06 <?php
07 if(isset($_POST['ok_num']))
08 {
09     $num=$_POST['txtnum'];
10     echo "<hr />";
11     echo "Ban dang chon $num file upload<br />";
12     echo "<form action='doupload.php?file=$num' method='post'
13 enctype='multipart/form-data'>";
14     for($i=1; $i <= $num; $i++)
15     {
16         echo "<input type='file' name='img[]' /><br />";
17     }
18     echo "<input type='submit' name='ok_upload'
19 value='Upload' />";
20     echo "</form>";
21 }
```

Tại trang thực hiện nhiệm vụ doupload.php ta cần kiểm tra xem, người dùng có nhấn nút upload từ trang bên kia hay không ?. Nếu có ta sẽ xử lý công đoạn upload ấy, nếu không ta cần thông báo lỗi cho người dùng và yêu cầu họ chọn file.

```
01 <<?php
02 if(isset($_POST['ok_upload']))
03 {
04     //thực thi upload
05 }
06 else
```

```

07 {
08     echo "Vui long chon hinh truoc khi truy cap vao trang nay";
09 }
10 ?>

```

Như vậy, khi người dùng nhấn upload, ta sẽ lấy được tham số là biến \$num từ liên kết \$_GET['file'] mà ta đã thiết lập trên phần form action của file upload ở trang trước.

Biến \$num này dùng để thực thi việc upload và tuần tự lấy các thông tin của file.

Để upload file từ máy lên hệ thống ta cần sử dụng hàm:

`move_uploaded_file(tên_file_tạm,đường dẫn tới hệ thống).`

Có thể diễn giải nôm na công việc của hàm này như sau: Ngay sau khi người dùng nhấn upload, hệ thống sẽ tạo ra 5 tham số ứng với từng file. Cụ thể là tên tạm (tmp_name), tên gốc (name), kích thước (size), định dạng mime (type) và lỗi (error) nếu có. Và khác với kiểu nhập liệu thông thường, khi chúng ta sử dụng <input type=file name=ten> thì lúc này sẽ phát sinh một biến môi trường mới là \$_FILES['ten'] và dĩ nhiên nó sẽ đi kèm với 5 tham số cụ thể ở trên.

Ví dụ: \$_FILES['ten']['name'] //Lấy ra tên gốc của file.

Vậy ứng với cú pháp upload ở trên ta sẽ có:

`move_uploaded_file($_FILES['img']['tmp_name'], "data/".$_FILES['img']['name']);`

Cú pháp này sẽ tự động upload hình ảnh lên thư mục data trên hệ thống. Vậy ta cần phải tạo thư mục data ngang cấp với 2 file upload và doupload. Nếu là host thật, bạn cần CHMOD thư mục là 777 (cho phép thư mục có toàn quyền) thì mới thực thi upload file được.

Trong tình huống của chúng ta, vì là upload một lúc nhiều file, nên lúc này ta phải sử dụng tham số mảng tại trang upload ta đã khởi tạo.

Vậy ứng với biến \$num ta nhận từ upload ở trên ta sẽ triển khai code đơn giản như sau:

```

1 <?php
2 for($i=0; $i< $num; $i++)
3     {
4         move_uploaded_file($_FILES['img']['tmp_name'][$i], "data/".$_FILES['img']['name'][$i]);
5         $url="data/".$_FILES['img']['name'][$i];
6         $name=$_FILES['img']['name'][$i];
7     }
8 ?>

```

Với biến \$i là số lượng tăng lên theo vòng lặp. Vì mảng luôn bắt đầu bởi khóa là 0 nên ta cần phải thiết lập khởi tạo biến \$i với giá trị là 0. Biến \$url và \$name là đường dẫn hình ảnh và tên gốc của tấm hình được sử dụng để lưu vào CSDL.

Kế tới, ta cần lưu thông tin hình ảnh tương ứng vào cơ sở dữ liệu. Để làm được điều này, ta cần kết nối CSDL với cú pháp sau: (xem lại [bài 10: kết hợp php & mysql trong ứng dụng website](#)).

```
1 <?php
2 $conn=mysql_connect("localhost","root","root") or die("can't connect
  your database");
3 mysql_select_db("images",$conn);
4 ?>
```

Như vậy mỗi khi upload thành công một hình ảnh, ta phải thêm hình ảnh đó vào CSDL như sau:

```
1 <?php
2 $sql="insert into images(img_url,img_name) values('$url','$name')";
3 mysql_query($sql);
4 ?>
```

Sau khi thêm vào CSDL ta cần hiển thị thông tin và hình ảnh vừa upload để cho người dùng biết họ đã upload thành công.

Khi đó ta sử dụng một biến \$site để lưu đường dẫn gốc hiển thị trên textbox.

Demo Images Script - Copyright by QHOnline.Info

Upload Thanh cong file **akira.jpg**



www.qhonline.info

Images URL: <http://www.qhonline.info/data/akira.jpg>

Upload Thanh cong file **fullhouse7.jpg**



www.qhonline.info

Images URL: <http://www.qhonline.info/data/fullhouse7.jp>

Upload Thanh cong file **fullhouse8.jpg**



www.qhonline.info

Images URL: <http://www.qhonline.info/data/fullhouse8.jp>

Vậy hoàn chỉnh code cho file doupload.php này như sau:

```
01 <?php
02 $site="http://www.qhonline.info";
03 if(isset($_POST['ok_upload']))
04 {
05     $num=$_GET['file'];
06     echo "<h3>Demo Images Script - Copyright by QHOnline.Info</h3>";
07     $conn=mysql_connect("localhost","root","root") or die("can't connect
your database");
08     mysql_select_db("images",$conn);
09     for($i=0; $i< $num; $i++)
10     {
11         move_uploaded_file($_FILES['img']['tmp_name'][$i],"data/".$_FILES['
12 img']['name'][$i]);
13         $url="data/".$_FILES['img']['name'][$i];
14         $name=$_FILES['img']['name'][$i];
15         $sql="insert into images(img_url,img_name) values('$url','$name)";
16         mysql_query($sql);
17         echo "Upload Thanh cong file <b>$name</b><br />";
18         echo "<img src='$url' width='120' /><br />";
19         echo "Images URL: <input type='text' name='link' value='$site/$url'
```

```

        size='35' /><br />";
19
20 }
21 mysql_close($conn);
22 }
23 else
24 {
25     echo "Vui long chon hinh truoc khi truy cap vao trang nay";
26 }
27 ?>

```

Download toàn bộ mã nguồn của bài này [tại đây](#).

Mọi sự góp ý, cũng như thắc mắc của các bạn dành cho bài viết vui lòng sử dụng tính năng [thảo luận tại diễn đàn](#). Hoặc thông qua việc [liên hệ của hệ thống website](#). Mong sẽ nhận được nhiều ý kiến đóng góp từ các bạn.

Bài 19: Viết ứng dụng tạo giỏ hàng - shopping cart - phần 1



Ở những bài trước, chúng ta đã đề cập tới các ứng dụng cơ bản, gần gũi với môi trường làm việc của chúng ta. Trong bài này, tôi sẽ tiếp tục hướng dẫn các bạn xây dựng hệ thống giỏ hàng (**shopping cart**) một ứng dụng phổ biến rất thường gặp trên các website cung cấp sản phẩm hiện nay. Nó gần như được thay thế cho việc mua sắm, việc lựa chọn thủ công mà chúng ta vẫn

thường làm.

Do nội dung và kiến thức của bài này khá dài, nên tôi sẽ trình bày bài viết này ở 2 phần, để giúp các bạn dễ theo dõi và dễ nắm bắt các kiến thức mà tôi đề cập về ứng dụng.

Phần 1: Xây dựng trang hiển thị sản phẩm.

Đầu tiên, ta khởi tạo bảng CSDL đơn giản như sau (xem lại [bài 9: cơ bản về ngôn ngữ sql và mysql](#))

```

1 CREATE TABLE `books` (
2   `id` int(10) unsigned NOT NULL auto_increment,
3   `title` varchar(255) NOT NULL,
4   `author` varchar(100) NOT NULL,
5   `price` int(30) NOT NULL,
6   PRIMARY KEY (`id`)
7 );

```

Ở trên là một bảng lưu thông tin của các quyển sách với tên, tác giả, giá tiền của các quyển sách.

```
1 INSERT INTO `books` VALUES (1, 'PHP Can Ban', 'Kenny', 115);
2 INSERT INTO `books` VALUES (2, 'PHP Nang Cao', 'Kenny', 150);
3 INSERT INTO `books` VALUES (3, 'PHP Framework', 'Kenny', 300);
4 INSERT INTO `books` VALUES (4, 'Joomla Can Ban', 'Kenny', 100);
```

Tiếp tục ta tạo file style.css để trình bày dữ liệu đẹp mắt

```
01 body{
02 font:12px verdana;
03 }
04 .pro{
05 border:1px solid #999999;
06 margin:5px;
07 padding:5px;
08 width:400px;
09 }
10 a{
11 color:#666666;
12 text-decoration:none;
13 font-weight:900;
14 }
15 #cart{
16 border:1px solid #999999;
17 margin:5px;
18 padding:5px;
19 width:400px;
20 text-align:center;
21 }
```

Sau khi đã hoàn tất việc chuẩn bị cơ sở dữ liệu và giao diện, tiếp theo ta sẽ xây dựng trang hiển thị các món hàng trên website, cho phép người sử dụng lựa chọn sách khi truy cập.

Để liệt kê danh sách các quyển sách đang có trong database, ta cần kết nối CSDL với thao tác code như sau. (xem lại [bài 10: kết hợp php & mysql trong ứng dụng website](#)).

```
1 <?php
2 $connect=mysql_connect("localhost","root","root")
3 or die("Can not connect database");
4 mysql_select_db("shop",$connect);
5 ?>
```

Lúc này, ta có thể liệt kê các quyển sách bằng cú pháp sau:

```
01 <?php
02 $sql="select * from books order by id desc";
```

```

03 $query=mysql_query($sql);
04 if(mysql_num_rows($query) > 0)
05 {
06   while($row=mysql_fetch_array($query))
07   {
08     echo "<div class='pro'>";
09     echo "<h3>$row[title]</h3>";
10     echo "Tac Gia: $row[author] - Gia: ".number_format($row[price],3)."
11     VND<br />";
12     echo "<p align='right'><a href='addcart.php?item=$row[id]'>Mua Sach
13     Nay</a></p>";
14     echo "</div>";
15   }
16 }
17 ?>

```

Đoạn code ở trên thực thi việc hiển thị sách nếu trong CSDL ít nhất 1 record. Và chúng sẽ liệt kê tiêu đề sách, tác giả, giá tiền. Ở đây, tôi sử dụng number_format() để lấy ra 3 số 000 cuối, ứng với đơn vị tiền tệ của Việt Nam là VNĐ.

Sau cùng tôi tạo một liên kết cho phép thêm quyền sách đó vào giỏ hàng nếu người mua cảm thấy ưng ý. (addcart.php sẽ là trang thêm vào giỏ hàng với tham số là mã của quyền sách).

Demo Shopping Cart

Ban khong co mon hang nao trong gio hang	
Joomla Layout Tac Gia: Bui Quoc Huy - Gia: 115.000 VND Mua Sach Nay	www.qhonline.info
Zend Framework Tac Gia: Bui Quoc Huy - Gia: 200.000 VND Mua Sach Nay	www.qhonline.info

Nói đến ứng dụng shopping cart thì hiện nay, trên thị trường tồn tại nhiều phương pháp code. Có thể sử dụng chuỗi để lưu giỏ hàng, cũng có thể lưu vào CSDL nháp giỏ hàng của người dùng và cũng có thể sử dụng mảng và session để lưu thông tin giỏ hàng. Trong khuôn khổ bài viết này, tôi sẽ sử dụng session và mảng để lưu thông tin giỏ hàng. (xem lại [Bài 7: Khái niệm cơ bản về Cookie và session trong PHP](#))

Tại trang addcart.php tôi cần khởi tạo một session và lưu mã quyền sách vào một mảng. Cụ thể là: `$_SESSION['cart'][$id]` (xem lại [bài 6: tổng quan về mảng về các hàm hỗ trợ](#)). Với `$id` là mã quyền sách mà người dùng đã chọn ở trang xem hàng hóa (`index.php`). Mục đích chính của trang addcart này là lưu trữ hoặc tính toán lại số lượng sản phẩm khi mà họ lựa chọn. (lưu ý là số lượng các món hàng).

Một vấn đề đặt ra trong trang này, là làm thế nào để nhận biết món hàng người đó đã chọn hay chưa. Chẳng hạn. Lần đầu tôi chọn mua quyền A, sau đó tôi quay lại chọn mua tiếp quyền A. vậy trong giỏ hàng phải ghi nhận số lượng quyền A này là 2. Chứ không thể chỉ lưu là 1 được.

Vậy, lúc này ta sẽ kiểm tra xem. Quyền sách mà ta vừa chọn có tồn tại trong giỏ hàng hay chưa. Nếu có, ta phải tiến hành lấy số lượng đang có tăng lên 1 đơn vị. Còn nếu không, ta phải gán số lượng của chúng là 1.

Code xử lý hoàn chỉnh trang addcart.php này sẽ như sau:

```
01 <?php
02 session_start();
03 $id=$_GET['item'];
04 if(isset($_SESSION['cart'][$id]))
05 {
06     $qty = $_SESSION['cart'][$id] + 1;
07 }
08 else
09 {
10     $qty=1;
11 }
12 $_SESSION['cart'][$id]=$qty;
13 header("location:cart.php");
14 exit();
15 ?>
```

Nôm na, chúng ta có thể hiểu addcart chỉ đơn giản là xử lý số lượng hàng hóa và lưu chúng ở dạng mảng mà thôi.

Như vậy tại trang mua sách, ta cũng cần cho khách hàng biết rằng trong giỏ hàng của họ hiện đang có bao nhiêu món hàng. Hoặc nếu chưa có món nào, ta cũng phải báo cho họ biết về việc đó.

Vậy khi nào thì giỏ hàng rỗng ?. Đó là khi session của giỏ hàng không tồn tại Id của quyền sách nào. Cụ thể, `$_SESSION['cart'][$id]`. Khi id không tồn tại trong session này thì cũng là lúc giỏ hàng không tồn tại.

Vậy trước khi cho hiển thị giỏ hàng, ta cần kiểm tra xem có tồn tại id nào trong giỏ hàng hay không. Và vì id lưu ở dạng mảng đa chiều, nên ta cần dùng vòng lặp duyệt mảng foreach.

foreach(\$_SESSION['cart'] as \$k=>\$v)

Với \$k có ý nghĩa tương đương \$id quyền sách và \$v tương đương là số lượng của quyền sách trong giỏ hàng. Vậy nếu tồn tại biến \$k, thì tức có nghĩa là trong giỏ hàng có sách. Khi đó ta sử dụng một biến đã để báo hiệu rằng sách có tồn tại trong giỏ hàng hay không.

```
01 <?php
02 $ok=1;
03 if(isset($_SESSION['cart']))
04 {
05     foreach($_SESSION['cart'] as $k=>$v)
06     {
07         if(isset($k))
08         {
09             $ok=2;
10         }
11     }
12 }
13 if ($ok != 2)
14 {
15     echo '<p>Ban khong co mon hang nao trong gio hang</p>';
16 } else {
17     $items = $_SESSION['cart'];
18     echo '<p>Ban dang co <a href="cart.php">'.count($items).' mon hang
trong gio hang</a></p>';
19 }
20 ?>
```

Trong đoạn code trên ta sử dụng hàm count nhằm đếm xem trong mảng hiện tại (giỏ hàng) hiện đang có bao nhiêu quyền sách. Và thực thi việc báo cho người dùng biết họ đang có bao nhiêu quyền sách trong giỏ hàng.

Toàn bộ code xử lý của trang mua sách (index.php) này như sau:

```
01 <?php
02 session_start();
03 ?>
04 <html>
05 <head>
06 <title>Demo Shopping Cart - Created By My Kenny</title>
07 <link rel="stylesheet" href="style.css" />
08 </head>
09 <body>
```

```

10 <h1>Demo Shopping Cart</h1>
11 <div id='cart'>
12 <?php
13 $ok=1;
14 if(isset($_SESSION['cart']))
15 {
16     foreach($_SESSION['cart'] as $k=>$v)
17     {
18         if(isset($v))
19         {
20             $ok=2;
21         }
22     }
23 }
24 if ($ok != 2)
25 {
26     echo '<p>Ban khong co mon hang nao trong gio hang</p>';
27 } else {
28     $items = $_SESSION['cart'];
29     echo '<p>Ban dang co <a href="cart.php">'.count($items).' mon hang
    trong gio hang</a></p>';
30 }
31 ?>
32 </div>
33 <?php
34 $connect=mysql_connect("localhost","root","root")
35 or die("Can not connect database");
36 mysql_select_db("shop",$connect);
37 $sql="select * from books order by id desc";
38 $query=mysql_query($sql);
39 if(mysql_num_rows($query) > 0)
40 {
41     while($row=mysql_fetch_array($query))
42     {
43         echo "<div class=pro>";
44         echo "<h3>$row[title]</h3>";
45         echo "Tac Gia: $row[author] - Gia: ".number_format($row[price],3)."
    VND<br />";
46         echo "<p align='right'><a href='addcart.php?item=$row[id]'>Mua Sach
    Nay</a></p>";
47         echo "</div>";
48     }
49 }

```

50

51 ?>

52 </body>

53 </html>

Như vậy, trong bài này chúng ta đã hoàn tất trang xem sách và trang xử lý việc thêm sản phẩm mới vào giỏ hàng như thế nào. Mời các bạn đọc tiếp bài sau với tính năng quản lý, sửa, xóa từng món hàng hoặc cả giỏ hàng trong hệ thống.

Bài 20: Viết ứng dụng tạo giỏ hàng - shopping cart - phần 2



Ở phần 1 tôi đã trình bày phần hiển thị thông tin sách có trong cơ sở dữ liệu, chu trình thực thi của việc thêm một quyển sách vào giỏ hàng. Và cơ chế quản lý giỏ hàng của chúng ta như thế nào. Tiếp tục với phần 2, tôi sẽ trình bày cách quản lý giỏ hàng như chỉnh sửa món hàng, liệt kê hàng hóa trong giỏ hàng và xóa bỏ từng món hàng hay cả giỏ hàng như thế nào. Để có thể hiểu được

phần 2. Các bạn vui lòng xem lại

Phần 2: Xây dựng hệ thống quản lý giỏ hàng.

Sau khi đã thêm một món hàng, tại giỏ hàng ta cũng cần phải xử lý để hiển thị các món hàng đã có trong giỏ. Và như phương pháp ở phần 1, chúng ta tạm lưu mã sách trong session. (xem lại [Bài 7: Khái niệm cơ bản về Cookie và session trong PHP](#)). Vậy ở trang giỏ hàng này, ta sẽ khởi tạo session và sử dụng vòng lặp duyệt mảng foreach để lặp toàn bộ mã sách đang lưu trong session. Tiếp tục, ta lại chuyển chúng sang dạng chuỗi bằng hàm implode().

```
1 <?php
2 session_start();
3 foreach($_SESSION['cart'] as $key=>$value)
4 {
5     $item[]=$key;
6 }
7 $str=implode(",",$item);
8 ?>
```

Giả sử lúc này chuỗi của chúng ta sẽ có dạng 7,8,9. Công việc tiếp theo là kết nối CSDL để liệt kê các sản phẩm có mã như ở trên. (xem lại [bài 10: kết hợp php & mysql trong ứng dụng website](#)). Thay vì sử dụng select * from tên_bảng where id= ?? . Thì để tối ưu hơn, tôi sẽ sử dụng phép in trong SQL. Lúc này câu truy vấn sẽ tương đương:

```
1 <?php
2 $sql="select * from books where id in ('$str') order by id desc";
```

```

3 $query=mysql_query($sql);
4 while($row=mysql_fetch_array($query))
5 ?>

```

Tiếp tục, ta lập toàn bộ thông tin sách bao gồm tên, tác giả, giá tiền và cả số lượng mà ta đã lưu trong session là \$_SESSION['cart'][\$ID_Món_Hàng]. ID_Món_hàng chính là thông tin ta lập ra từ CSDL (\$row[id]).

Bên cạnh đó tại phần số lượng, ta sẽ đưa giá trị lưu ở session ra textbox (vì tại giỏ hàng, người dùng được phép điều chỉnh số lượng, nên lúc này ta cần tạo textbox cho họ điều chỉnh). Vì lập toàn bộ các quyền sách nên tại tên của textbox số lượng ta cũng cần truyền id để nhận biết số lượng đó thuộc ID của quyền sách nào.

```

1 <?php
  echo "<p align='right'>So Luong: <input type='text'
2 name='qty[$row[id]]' size='5' value='{$_SESSION['cart'][$row[id]]}' />
  - ";
3 ?>

```

Chúng ta cũng cho người dùng được phép xóa 1 món hàng nào đó ra khỏi giỏ hàng của họ. Bằng cách truyền mã quyền sách của từng quyền vào liên kết delcart.php.

```

1 <?php
2 echo "<a href='delcart.php?productid=$row[id]'>Xoa Sach Nay</a></p>";
3 ?>

```

Tại đây, ta cũng cần tính luôn giá tiền của từng quyền sách tương ứng với số lượng mà họ đã chọn. Như vậy, số lượng là phần ta lưu ở session, còn giá tiền là phần ta lấy ra từ CSDL ứng với mảng \$row (\$row['price']).

```

1 <?php
2 echo "<p align='right'> Gia tien cho mon hang: ".
  number_format($_SESSION['cart'][$row[id]]*$row[price],3) ." VND</p>";
3 ?>

```

Sau cùng, ta cần tính tổng tiền của toàn bộ sản phẩm có trong giỏ hàng. Bằng cách cộng dồn tổng giá tiền của từng món.

```

1 <?php
2 $total+=$_SESSION['cart'][$row[id]]*$row[price];
3 ?>

```

Phần còn lại, là chúng ta hiển thị giá tiền với đúng định dạng VND của Việt Nam.

```

1 <?php
2 echo "<b>Tong tien cho cac mon hang: <font color='red'>".
  number_format($total,3) ." VND</font></b>";
3 ?>

```

Đồng thời, ta cũng tạo nút cho phép người dùng cập nhật. Và cho phép người dùng xóa toàn bộ giỏ hàng.

```

1 <?php
2 echo "<input type='submit' name='submit' value='Cap Nhat Gio Hang'
  />";

```

```

3 echo "<div class='pro' align='center'>";
4 echo "<b><a href='index.php'>Mua Sach Tiep</a> - <a
4 href='delcart.php?productid=0'>Xoa Bo Gio Hang</a></b>";
5 ?>

```

Như vậy, code đầy đủ sẽ là:

```

01 <?php
02 echo "<form action='cart.php' method='post'>";
03 foreach($_SESSION['cart'] as $key=>$value)
04 {
05     $item[]=$key;
06 }
07 $str=implode(",",$item);
08 $connect=mysql_connect("localhost","root","root") or die("Can not
connect database");
09 mysql_select_db("shop",$connect);
10 $sql="select * from books where id in ($str)";
11 $query=mysql_query($sql);
12 while($row=mysql_fetch_array($query))
13 {
14     echo "<div class='pro'>";
15     echo "<h3>$row[title]</h3>";
16     echo "Tac gia: $row[author] - Gia: ".number_format($row[price],3)."
VND<br />";
17     echo "<p align='right'>So Luong: <input type='text'
name='qty[$row[id]]' size='5' value='{$_SESSION['cart'][$row[id]]}'>
- ";
18     echo "<a href='delcart.php?productid=$row[id]'>Xoa Sach
Nay</a></p>";
19     echo "<p align='right'> Gia tien cho mon hang: ".
number_format($_SESSION['cart'][$row[id]]*$row[price],3) ." VND</p>";
20     echo "</div>";
21     $total+=$_SESSION['cart'][$row[id]]*$row[price];
22 }
23 echo "<div class='pro' align='right'>";
24 echo "<b>Tong tien cho cac mon hang: <font color='red'>".
number_format($total,3) ." VND</font></b>";
25 echo "</div>";
26 echo "<input type='submit' name='submit' value='Cap Nhat Gio Hang'
/>";
27 echo "<div class='pro' align='center'>";
28 echo "<b><a href='index.php'>Mua Sach Tiep</a> - <a
href='delcart.php?productid=0'>Xoa Bo Gio Hang</a></b>";
29 echo "</div>";
30 ?>

```

Sau khi thiết lập thành công trang giỏ hàng cơ bản, lúc này ta đã có thể thêm sách một cách dễ dàng. Tuy nhiên, giả sử trong trường hợp không có sách thì sao ?. Chúng ta vẫn chưa xét đến trường hợp giỏ hàng rỗng thì sẽ như thế nào. Vậy khi nào thì giỏ hàng rỗng ?. Đó là khi session của giỏ hàng không tồn tại Id của quyển sách nào. Cụ thể, \$_SESSION['cart'][id]. Khi id không tồn tại trong session này thì cũng là lúc giỏ hàng không tồn tại.

Demo Shopping Cart

Zend Framework Tác giả: Bui Quoc Huy - Giá: 200.000 VND Số Lượng: <input type="text" value="4"/> - Xóa Sách Nay Giá tiền cho món hàng: 800.000 VND	www.qhonline.info
Joomla Layout Tác giả: Bui Quoc Huy - Giá: 115.000 VND Số Lượng: <input type="text" value="3"/> - Xóa Sách Nay Giá tiền cho món hàng: 345.000 VND	www.qhonline.info
Tổng tiền cho các món hàng: 1,145.000 VND	
<input type="button" value="Cập Nhật Giỏ Hàng"/>	
<input type="button" value="Mua Sách Tiếp - Xóa Bỏ Giỏ Hàng"/>	

Vậy trước khi cho hiển thị giỏ hàng, ta cần kiểm tra xem có tồn tại id nào trong giỏ hàng hay không. Và vì id lưu ở dạng mảng đa chiều, nên ta cần dùng vòng lặp duyệt mảng foreach.

```
foreach($_SESSION['cart'] as $k=>$v)
```

Với \$k có ý nghĩa tương đương \$id quyển sách và \$v tương đương là số lượng của quyển sách trong giỏ hàng. Vậy nếu tồn tại biến \$k, thì tức có nghĩa là trong giỏ hàng có sách.

```
01 <?php
02 $ok=1;
03 if(isset($_SESSION['cart']))
04 {
05     foreach($_SESSION['cart'] as $k => $v)
06     {
07         if(isset($k))
```

```

08 {
09     $ok=2;
10 }
11 }
12 }
13 if($ok == 2)
14 {
15 // code xử lý giỏ hàng ở trên.
16 }
17 ?>

```

Đoạn code này, ta sử dụng biến \$ok để làm biến kiểm tra, mặc định khi load dữ liệu biến \$ok sẽ bằng 1. Và khi trong giỏ hàng tồn tại sách thì chúng ta sẽ thay đổi biến \$ok thành 2. Và gọi giỏ hàng như code ở trên.

Ở phần trên, ta cũng có đề cập khi người dùng tiến hành chỉnh sửa số lượng từng món hàng đơn lẻ qua textbox và nhấn cập nhật thì hệ thống sẽ tiến hành chỉnh sửa lại thông tin giỏ hàng. Vậy chúng ta sẽ xử lý như thế nào cho tương hợp đó.

Nếu chú ý, các bạn sẽ thấy dòng code chứa textbox cho phép người dùng nhập số lượng có một tham số đặc biệt là name=qty[\$row[id]]. Vậy tham số này được dùng để làm gì ?.

Tham số này, nói cho chúng ta biết số lượng đang hiển thị là thuộc mã sản phẩm nào. Vậy khi tiến hành cập nhật giỏ hàng ta sẽ kiểm tra, nếu \$qty[\$row[id]] mà có giá trị là 0. Tức là người đó muốn xóa bỏ giỏ hàng. Ngược lại, ta chỉ việc cập nhật giỏ hàng \$id tương ứng với số lượng nhập ở textbox.

```

01 <?php
02 if(isset($_POST['submit']))
03 {
04     foreach($_POST['qty'] as $key=>$value)
05     {
06         if( ($value == 0) and (is_numeric($value)))
07         {
08             unset ($_SESSION['cart'][$key]);
09         }
10         elseif(($value > 0) and (is_numeric($value)))
11         {
12             $_SESSION['cart'][$key]=$value;
13         }
14     }
15     header("location:cart.php");
16 }
17 ?>

```

Việc xóa món hàng ở trên chỉ đơn giản là hủy bỏ session của id đó. Việc cập nhật số lượng chỉ đơn giản là gán đè số lượng người nhập (thẻ value trong textbox) vào số lượng đang lưu trong session.

Vậy code hoàn chỉnh của trang cart.php này sẽ như sau:

```
01 <?php
02 session_start();
03 if(isset($_POST['submit']))
04 {
05     foreach($_POST['qty'] as $key=>$value)
06     {
07         if( ($value == 0) and (is_numeric($value)))
08         {
09             unset ($_SESSION['cart'][$key]);
10         }
11         elseif(($value > 0) and (is_numeric($value)))
12         {
13             $_SESSION['cart'][$key]=$value;
14         }
15     }
16     header("location:cart.php");
17 }
18 ?>
19 <html>
20 <head>
21 <title>Demo Shopping Cart - Created By My Kenny</title>
22 <link rel="stylesheet" href="style.css" />
23 </head>
24 <body>
25 <h1>Demo Shopping Cart</h1>
26 <?
27 $ok=1;
28 if(isset($_SESSION['cart']))
29 {
30     foreach($_SESSION['cart'] as $k => $v)
31     {
32         if(isset($k))
33         {
34             $ok=2;
35         }
36     }
37 }
38 if($ok == 2)
```

```

39 {
40
41     echo "<form action='cart.php' method='post'>";
42     foreach($_SESSION['cart'] as $key=>$value)
43     {
44         $item[]=$key;
45     }
46     $str=implode(",",$item);
47     $connect=mysql_connect("localhost","root","root") or die("Can not
connect database");
48     mysql_select_db("shop",$connect);
49     $sql="select * from books where id in ($str)";
50     $query=mysql_query($sql);
51     while($row=mysql_fetch_array($query))
52     {
53         echo "<div class='pro'>";
54         echo "<h3>$row[title]</h3>";
55         echo "Tac gia: $row[author] - Gia:
".number_format($row[price],3)." VND<br />";
56         echo "<p align='right'>So Luong: <input type='text'
name='qty[$row[id]]' size='5' value='{$_SESSION['cart'][$row[id]]}'>
- ";
57         echo "<a href='delcart.php?productid=$row[id]'>Xoa Sach
Nay</a></p>";
58         echo "<p align='right'> Gia tien cho mon hang: ".
number_format($_SESSION['cart'][$row[id]]*$row[price],3)." VND</p>";
59         echo "</div>";
60         $total+=$_SESSION['cart'][$row[id]]*$row[price];
61     }
62     echo "<div class='pro' align='right'>";
63     echo "<b>Tong tien cho cac mon hang: <font color='red'>".
number_format($total,3)." VND</font></b>";
64     echo "</div>";
65     echo "<input type='submit' name='submit' value='Cap Nhat Gio
Hang'>";
66     echo "<div class='pro' align='center'>";
67     echo "<b><a href='index.php'>Mua Sach Tiep</a> - <a
href='delcart.php?productid=0'>Xoa Bo Gio Hang</a></b>";
68     echo "</div>";
69 }
70 else
71 {
72     echo "<div class='pro'>";
73     echo "<p align='center'>Ban khong co mon hang nao trong gio hang<br
/><a href='index.php'>Buy Ebook</a></p>";

```

```

74     echo "</div>";
75 }
76 ?>
77 </body>
78 </html>

```

Và cuối cùng, khi người dùng nhấn xóa toàn bộ giỏ hàng hay chỉ xóa một món hàng, ta sẽ gọi tới trang delcart.php. Vậy trang này sẽ xử lý như thế nào ?.

Tại đây, ta có thể nhận tham số là: `$_GET['productid'];`

Tham số này sẽ là \$id mà chúng truyền qua liên kết. Nếu xóa toàn bộ giỏ hàng, tức ta sẽ truyền cho nó giá trị bằng 0. Lúc này, ta sẽ hủy toàn bộ `$_SESSION['cart']`. Ngược lại, nếu là một \$id cụ thể, thì ta chỉ xóa món hàng đó mà thôi. `$_SESSION['cart'][$id]`.

Code hoàn chỉnh của file delcart.php như sau:

```

01 <?php
02 session_start();
03 $cart=$_SESSION['cart'];
04 $id=$_GET['productid'];
05 if($id == 0)
06 {
07     unset($_SESSION['cart']);
08 }
09 else
10 {
11     unset($_SESSION['cart'][$id]);
12 }
13 header("location:cart.php");
14 exit();
15 ?>

```

Download toàn bộ mã nguồn của bài này [tại đây](#).

Như vậy, chúng ta đã hoàn tất việc xây dựng hoàn chỉnh một hệ thống **shopping cart** đơn giản. Tuy rằng, đây không phải là một bài viết hoàn chỉnh trong việc xây dựng mô hình **thương mại điện tử**. Nhưng quá đó, phần nào giúp các bạn hiểu và dễ dàng phát triển hệ thống của mình có tổ chức hơn.

Hy vọng trong thời gian tới, tôi sẽ viết tiếp bài viết sử dụng mô hình **thanh toán trực tuyến (paypal)** kết hợp với giỏ hàng để hoàn chỉnh bài hướng dẫn về **shopping cart** này. Mọi sự góp ý, cũng như thắc mắc của các bạn dành cho bài viết vui lòng sử dụng tính năng **thảo luận tại diễn đàn**. Hoặc thông qua việc trả lời với khung **phản hồi bên dưới**. Mong sẽ nhận được nhiều ý kiến đóng góp từ các bạn.

Chapter 30 : Các hàm

PHP 5 mảng chức năng

Các mảng chức năng PHP là một phần của lõi PHP. Không cần cài đặt để sử dụng các chức năng này.

Chức năng	Mô tả
<u>array ()</u>	Tạo ra một mảng
<u>array_change_key_case ()</u>	Thay đổi tất cả các phím trong một mảng chữ thường hoặc chữ hoa
<u>array_chunk ()</u>	Chia một mảng thành nhiều phần của mảng
<u>array_combine ()</u>	Tạo ra một mảng bằng cách sử dụng các yếu tố từ một mảng "chìa khóa" và m mảng "giá trị"
<u>array_count_values ()</u>	Trả về một mảng với số lần xuất hiện cho mỗi giá trị
<u>array_diff ()</u>	So sánh giá trị mảng, và trả về sự khác biệt
<u>array_diff_assoc ()</u>	So sánh mảng và giá trị, và trả lại sự khác biệt
<u>array_diff_key ()</u>	So sánh trong mảng, và trả lại sự khác biệt
<u>array_diff_uassoc ()</u>	So sánh mảng và giá trị, với một kiểm tra chức năng bổ sung người dùng tạo, v trả lại sự khác biệt
<u>array_diff_ukey ()</u>	So sánh các khóa mảng, với một kiểm tra chức năng bổ sung người dùng tạo, v trả lại sự khác biệt
<u>array_fill ()</u>	Điền vào một mảng với giá trị
<u>array_fill_keys ()</u>	Điền vào một mảng với giá trị, chỉ định phím
<u>array_filter ()</u>	Bộ lọc các yếu tố của một mảng bằng cách sử dụng một chức năng người dùng tạo ra
<u>array_flip ()</u>	Trao đổi tất cả các phím với các giá trị liên quan của họ trong một mảng
<u>array_intersect ()</u>	So sánh giá trị mảng, và trả về các trận đấu
<u>array_intersect_assoc ()</u>	So sánh mảng và giá trị, và trả về các trận đấu
<u>array_intersect_key ()</u>	So sánh trong mảng, và trả về các trận đấu

<u>array_intersect_uassoc()</u>	So sánh mảng và giá trị, với một kiểm tra chức năng bổ sung người dùng tạo, và trả về các trận đấu
<u>array_intersect_ukey()</u>	So sánh các khóa mảng, với một kiểm tra chức năng bổ sung người dùng tạo, và trả về các trận đấu
<u>array_key_exists()</u>	Kiểm tra nếu phím quy định tồn tại trong mảng
<u>array_keys()</u>	Trả về tất cả các phím của một mảng
<u>array_map()</u>	Gửi mỗi giá trị của một mảng một chức năng người sử dụng thực hiện, trong đó trả về các giá trị mới
<u>array_merge()</u>	Kết hợp một hoặc nhiều mảng thành một mảng
<u>array_merge_recursive()</u>	Kết hợp một hoặc nhiều mảng thành một mảng
<u>array_multisort()</u>	Sắp xếp nhiều hoặc mảng đa chiều
<u>array_pad()</u>	Chèn một số quy định của các mục, với một giá trị chỉ định, vào một mảng
<u>array_pop()</u>	Xóa phần tử cuối cùng của mảng
<u>array_product()</u>	Tính sản phẩm của các giá trị trong một mảng
<u>array_push()</u>	Chèn một hoặc nhiều yếu tố để kết thúc của một mảng
<u>array_rand()</u>	Trả về một hoặc nhiều phím ngẫu nhiên từ một mảng
<u>array_reduce()</u>	Trả về một mảng như một chuỗi, bằng cách sử dụng một chức năng người dùng định nghĩa
<u>array_replace()</u>	Thay thế các yếu tố từ mảng thông qua vào mảng đầu tiên
<u>array_replace_recursive()</u>	Thay thế các yếu tố từ các mảng thông qua vào mảng đầu tiên đệ quy
<u>array_reverse()</u>	Trả về một mảng theo thứ tự đảo ngược
<u>array_search()</u>	Tìm kiếm một mảng cho một giá trị nhất định và trả lại chìa khóa
<u>array_shift()</u>	Loại bỏ các yếu tố đầu tiên từ một mảng và trả về giá trị của phần tử bị loại bỏ
<u>array_slice()</u>	Trả về lựa chọn các bộ phận của một mảng
<u>array_splice()</u>	Loại bỏ và thay thế các yếu tố quy định của một mảng
<u>array_sum()</u>	Trả về tổng các giá trị trong một mảng
<u>array_udiff()</u>	So sánh các giá trị mảng trong một chức năng người sử dụng tạo và trả về một mảng
<u>array_udiff_assoc()</u>	So sánh các khóa mảng, so sánh giá trị mảng trong một chức năng người dùng tạo, và trả về một mảng

<u>array_udiff_uassoc()</u>	So sánh mảng và giá trị mảng trong các chức năng do người sử dụng tạo ra, và trả về một mảng
<u>array_uintersect()</u>	So sánh các giá trị mảng trong một chức năng người sử dụng tạo và trả về một mảng
<u>array_uintersect_assoc()</u>	So sánh các khóa mảng, so sánh giá trị mảng trong một chức năng người dùng tạo, và trả về một mảng
<u>array_uintersect_uassoc()</u>	So sánh mảng và giá trị mảng trong các chức năng do người sử dụng tạo ra, và trả về một mảng
<u>array_unique()</u>	Loại bỏ các giá trị nhân bản từ một mảng
<u>array_unshift()</u>	Thêm một hoặc nhiều yếu tố để bắt đầu của một mảng
<u>array_values()</u>	Trả về tất cả các giá trị của một mảng
<u>array_walk()</u>	Áp dụng một chức năng người sử dụng tất cả các thành viên của một mảng
<u>array_walk_recursive()</u>	Áp dụng một chức năng người dùng đệ quy để tất cả các thành viên của một mảng
<u>arsort()</u>	Sắp xếp một mảng kết hợp thứ tự giảm dần theo giá trị
<u>asort()</u>	Sắp xếp một mảng kết hợp theo thứ tự tăng dần theo giá trị
<u>compact()</u>	Tạo ra các biến mảng có chứa và các giá trị của họ
<u>count()</u>	Trả về số phần tử trong một mảng
<u>current()</u>	Trả về phần tử hiện tại trong một mảng
<u>each()</u>	Trả về các khóa hiện tại và cặp giá trị từ một mảng
<u>end()</u>	Đặt con trỏ nội bộ của một mảng phần tử cuối cùng của nó
<u>extract()</u>	Nhập khẩu các biến vào bảng biểu tượng hiện tại từ một mảng
<u>in_array()</u>	Kiểm tra nếu một giá trị được chỉ định tồn tại trong một mảng
<u>key()</u>	Lấy một chìa khóa từ một mảng
<u>krsort()</u>	Sắp xếp một mảng kết hợp thứ tự giảm dần, theo chìa khóa
<u>ksort()</u>	Sắp xếp một mảng kết hợp trong thứ tự tăng dần, theo chìa khóa
<u>list()</u>	Gán biến như thể họ là một mảng
<u>natcasesort()</u>	Sắp xếp một mảng bằng cách sử dụng một trường hợp insensitive "trật tự tự nhiên" thuật toán
<u>natsort()</u>	Sắp xếp một mảng bằng cách sử dụng một thuật toán "trật tự tự nhiên"

<u>next ()</u>	Trước con trỏ mảng nội bộ của một mảng
<u>pos ()</u>	Bí danh hiện nay ()
<u>prev ()</u>	Tua lại con trỏ mảng nội bộ
<u>range ()</u>	Tạo ra một mảng có chứa một loạt các yếu tố
<u>reset ()</u>	Đặt con trỏ nội bộ của một mảng phần tử đầu tiên của nó
<u>rsort ()</u>	Sắp xếp một mảng lập chỉ mục thứ tự giảm dần
<u>shuffle ()</u>	Shuffles một mảng
<u>sizeof ()</u>	Bí danh của count ()
<u>sort ()</u>	Sắp xếp một mảng lập chỉ mục theo thứ tự tăng dần
<u>uasort ()</u>	Sắp xếp một mảng giá trị bằng cách sử dụng người dùng xác định chức năng so sánh
<u>uksort ()</u>	Sắp xếp một mảng các phím bằng cách sử dụng người dùng xác định chức năng so sánh
<u>usort ()</u>	Sắp xếp một mảng bằng cách sử dụng người dùng xác định chức năng so sánh

PHP Lịch Chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	PHP
<u>cal_days_in_month ()</u>	Trả về số ngày trong một tháng cho một năm quy định và lịch	4
<u>cal_from_jd ()</u>	Chuyển đổi một số ngày Julian thành một ngày lịch quy định	4
<u>cal_info ()</u>	Trả về thông tin về một lịch nhất định	4
<u>cal_to_jd ()</u>	Chuyển đổi một ngày để đếm Julian ngày	4
<u>easter_date ()</u>	Trả về dấu thời gian Unix nửa đêm ngày lễ Phục sinh của một năm quy định	3
<u>easter_days ()</u>	Trả về số ngày sau ngày 21 tháng ba, Lễ Phục Sinh rơi cho một năm quy định	3
<u>FrenchToJD ()</u>	Chuyển đổi một ngày của đảng Cộng hòa Pháp một số ngày Julian	3
<u>GregorianToJD ()</u>	Chuyển đổi một ngày Gregorian một số ngày Julian	3
<u>JDDayOfWeek ()</u>	Trả về ngày một tuần	3

<u>JDMonthName ()</u>	Trả về một tên tháng	3
<u>JDFrench ()</u>	Chuyển đổi một số ngày Julian đến một ngày đảng Cộng hòa Pháp	3
<u>JDTogregorian ()</u>	Chuyển đổi một số ngày Julian đến một ngày dương lịch	3
<u>jdtojewish ()</u>	Chuyển đổi một số ngày Julian đến một ngày người Do Thái	3
<u>JDT Julian ()</u>	Chuyển đổi một số ngày Julian đến một ngày Julian	3
<u>jdtonix ()</u>	Chuyển đổi một số ngày Julian một dấu thời gian Unix	4
<u>JewishToJD ()</u>	Chuyển đổi một ngày của người Do Thái một số ngày Julian	3
<u>JulianToJD ()</u>	Chuyển đổi một ngày Julian một số ngày Julian	3
<u>unixtojd ()</u>	Chuyển đổi một dấu thời gian Unix một số ngày Julian	4

PHP Lịch hằng số

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ các hằng số.

Liên tục	Mô tả	P
CAL_GREGORIAN	Lịch Gregory	3
CAL_JULIAN	Lịch Julian	3
CAL_JEWISH	Do Thái lịch	3
CAL_FRENCH	Cộng hòa lịch Pháp	3
CAL_NUM_CALS		3
CAL_DOW_DAYNO		3
CAL_DOW_SHORT		3
CAL_DOW_LONG		3
CAL_MONTH_GREGORIAN_SHORT		3
CAL_MONTH_GREGORIAN_LONG		3
CAL_MONTH_JULIAN_SHORT		3
CAL_MONTH_JULIAN_LONG		3
CAL_MONTH_JEWISH		3
CAL_MONTH_FRENCH		3

CAL_EASTER_DEFAULT		4
CAL_EASTER_ROMAN		4
CAL_EASTER_ALWAYS_GREGORIAN		4
CAL_EASTER_ALWAYS_JULIAN		4
CAL_JEWISH_ADD_ALAFIM_GERESH		5
CAL_JEWISH_ADD_ALAFIM		5
CAL_JEWISH_ADD_GERESHAYIM		5

Thời gian chạy cấu hình

Các hành vi của các chức năng ngày / thời gian bị ảnh hưởng bởi các thiết lập trong php.ini.

Ngày / Thời gian tùy chọn cấu hình:

Tên	Mặc định	Mô tả	Thay đổi
date.default_latitude	"31,7667"	Chỉ định vĩ độ mặc định (có sẵn từ PHP 5). Tùy chọn này được sử dụng bởi date_sunrise () và date_sunset ()	PHP_INI_ALL
date.default_longitude	"35,2333"	Chỉ định kinh độ mặc định (có sẵn từ PHP 5). Tùy chọn này được sử dụng bởi date_sunrise () và date_sunset ()	PHP_INI_ALL
date.sunrise_zenith	"90,83"	Chỉ định đỉnh cao của mặt trời mọc mặc định (có sẵn từ PHP 5). Tùy chọn này được sử dụng bởi date_sunrise () và date_sunset ()	PHP_INI_ALL
date.sunset_zenith	"90,83"	Chỉ định mặc định hoàng hôn đỉnh cao (có sẵn từ PHP 5). Tùy chọn này được sử dụng bởi date_sunrise () và date_sunset ()	PHP_INI_ALL
date.timezone	""	Chỉ định múi giờ mặc định (có sẵn từ PHP 5.1)	PHP_INI_ALL

PHP Ngày / Giờ Chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
<u>checkdate ()</u>	Xác nhận một ngày dương lịch	3
<u>date_default_timezone_get ()</u>	Trả về múi giờ mặc định	5

<u>date_default_timezone_set ()</u>	Thiết lập múi giờ mặc định	5
<u>date_sunrise ()</u>	Trả về thời gian mặt trời mọc cho một ngày nhất định / vị trí	5
<u>date_sunset ()</u>	Trả về thời gian của hoàng hôn cho một ngày nhất / vị trí	5
<u>date ()</u>	Định dạng thời gian địa phương / ngày	3
<u>getdate ()</u>	Trả về một mảng chứa thông tin ngày tháng và thời gian cho một dấu thời gian Unix	3
<u>gettimeofday ()</u>	Trả về một mảng chứa thông tin thời gian hiện tại	3
<u>gmdate ()</u>	Định dạng một ngày / giờ GMT / UTC	3
<u>gmmktime ()</u>	Trả về dấu thời gian Unix cho một ngày GMT	3
<u>gmstrftime ()</u>	Định dạng một thời gian GMT / UTC / ngày theo lập miền địa phương	3
<u>Idate ()</u>	Định dạng một thời gian địa phương / ngày là số nguyên	5
<u>localtime ()</u>	Trả về một mảng có chứa các thành phần thời gian của một dấu thời gian Unix	4
<u>microtime ()</u>	Trả về micro giây cho thời gian hiện tại	3
<u>mktime ()</u>	Trả về dấu thời gian Unix cho một ngày	3
<u>strftime ()</u>	Định dạng thời gian địa phương / ngày theo lập miền địa phương	3
<u>strtotime ()</u>	Phân tích một ngày thời gian / tạo ra với strftime ()	5
<u>strtotime ()</u>	Phân tích một văn bản tiếng Anh ngày hoặc thời gian vào một dấu thời gian Unix	3
<u>thời gian ()</u>	Trả về thời gian hiện tại như là một dấu thời gian Unix	3

PHP Ngày / Giờ hằng số

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ các hằng số.

Liên tục	Mô tả	P
DATE_ATOM	Atom (ví dụ: 2005-08-15T16: 13:03 0000)	
DATE_COOKIE	HTTP Cookies: (ví dụ: Sun, 14 Aug 2005 16:13:03 UTC)	
DATE_ISO8601	ISO-8601 (ví dụ: 2005-08-14T16: 13:03 0000)	

DATE_RFC822	RFC 822 (ví dụ: Sun, 14 Aug 2005 16:13:03 UTC)
DATE_RFC850	RFC 850 (ví dụ: Chủ nhật, 14-Aug-05 16:13:03 UTC)
DATE_RFC1036	RFC 1036 (ví dụ: Chủ nhật, 14-Aug-05 16:13:03 UTC)
DATE_RFC1123	RFC 1123 (ví dụ: Sun, 14 Aug 2005 16:13:03 UTC)
DATE_RFC2822	RFC 2822 (Sun, 14 Aug 2005 16:13:03 0000)
DATE_RSS	RSS (Sun, 14 Aug 2005 16:13:03 UTC)
DATE_W3C	World Wide Web Consortium (ví dụ: 2005-08-14T16: 13:03 0000)

PHP mục Chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
<u>chdir ()</u>	Thay đổi thư mục hiện tại	3
<u>chroot ()</u>	Thay đổi thư mục gốc của quá trình hiện tại	4
<u>dir ()</u>	Mở một xử lý thư mục và trả về một đối tượng	3
<u>closedir ()</u>	Đóng một xử lý thư mục	3
<u>getcwd ()</u>	Trả về thư mục hiện hành	4
<u>opendir ()</u>	Mở một xử lý thư mục	3
<u>readdir ()</u>	Trả về một mục từ một xử lý thư mục	3
<u>rewinddir ()</u>	Đặt lại một xử lý thư mục	3
<u>scandir ()</u>	Danh sách các tập tin và thư mục bên trong một đường dẫn cụ thể	5

PHP mục hằng số

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ các hằng số.

Liên tục	Mô tả	P
DIRECTORY_SEPARATOR		3

PATH_SEPARATOR

PHP Lỗi và chức năng đăng nhập

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
<u>debug_backtrace ()</u>	Tạo ra một backtrace	4
<u>debug_print_backtrace ()</u>	In một backtrace	5
<u>error_get_last ()</u>	Được các lỗi xảy ra lần cuối	5
<u>error_log ()</u>	Gửi một lỗi máy chủ đăng nhập vào một tập tin hoặc đến một địa điểm từ xa	4
<u>error_reporting ()</u>	Xác định lỗi được báo cáo	4
<u>restore_error_handler ()</u>	Phục hồi xử lý lỗi trước đó	4
<u>restore_exception_handler ()</u>	Phục hồi xử lý ngoại lệ trước đó	5
<u>set_error_handler ()</u>	Thiết lập một chức năng được xác định sử dụng để xử lý các lỗi	4
<u>set_exception_handler ()</u>	Thiết lập một chức năng được xác định sử dụng để xử lý các trường hợp ngoại lệ	5
<u>trigger_error ()</u>	Tạo ra một thông báo lỗi do người dùng định nghĩa	4
<u>user_error ()</u>	Bí danh của trigger_error ()	4

PHP Lỗi và hằng số Đăng nhập

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ các hằng số.

Giá trị	Liên tục	Mô tả	P
1	E_ERROR	Fatal lỗi thời gian chạy. Lỗi mà không thể được phục hồi từ.Thực thi kịch bản phải dừng lại	
2	E_WARNING	Lỗi thời gian chạy không gây tử vong. Thực thi kịch bản không dừng lại	
4	E_PARSE	Thời gian biên dịch lỗi phân tích cú pháp. Lỗi cú pháp chỉ nên được tạo ra bởi các cú pháp	

8	E_NOTICE	Thông báo thời gian chạy. Các kịch bản tìm thấy một cái gì đó có thể là một lỗi, nhưng cũng có thể xảy ra khi chạy một kịch bản bình thường	
16	E_CORE_ERROR	Fatal lỗi khởi động tại PHP. Điều này giống như một E_ERROR trong lỗi PHP	4
32	E_CORE_WARNING	Lỗi không gây tử vong tại PHP khởi động. Điều này giống như một E_WARNING trong lỗi PHP	4
64	E_COMPILE_ERROR	Fatal lỗi thời gian biên dịch. Điều này giống như một E_ERROR tạo ra bởi Zend Engine Scripting	4
128	E_COMPILE_WARNING	Lỗi thời gian biên dịch không gây tử vong. Điều này giống như một E_WARNING tạo ra bởi Zend Engine Scripting	4
256	E_USER_ERROR	Fatal người dùng tạo ra lỗi. Điều này giống như một bộ E_ERROR bằng cách lập trình bằng cách sử dụng trigger_error chức năng PHP ()	4
512	E_USER_WARNING	Không gây tử vong người dùng tạo ra cảnh báo. Điều này giống như một E_WARNING được thiết lập bởi các lập trình viên sử dụng PHP chức năng trigger_error ()	4
1024	E_USER_NOTICE	Người sử dụng tạo thông báo. Điều này giống như một E_NOTICE được thiết lập bởi các lập trình viên sử dụng PHP chức năng trigger_error ()	4
2048	E_STRICT	Thông báo thời gian chạy. PHP đề nghị thay đổi mã của bạn để giúp khả năng tương tác và khả năng tương thích của mã	5
4096	E_RECOVERABLE_ERROR	Lỗi catchable gây tử vong. Đây là như E_ERROR một nhưng có thể được đánh bắt bởi một xử lý người dùng định nghĩa (xem thêm set_error_handler ())	5
6143	E_ALL	Tất cả các lỗi và cảnh báo, ngoại trừ mức độ E_STRICT	5

Giới thiệu PHP hệ thống tập tin

Các chức năng hệ thống tập tin cho phép bạn truy cập và thao tác các hệ thống tập tin.

Cài đặt

Các chức năng hệ thống tập tin là một phần của lõi PHP. Không có cài đặt cần thiết để sử dụng các chức năng này.

Thời gian chạy cấu hình

Các hành vi của các chức năng hệ thống tập tin bị ảnh hưởng bởi các thiết lập trong php.ini.

Hệ thống tập tin cấu hình tùy chọn:

Tên	Mặc định	Mô tả	Thay đổi
allow_url_fopen	"1"	Cho phép fopen ()-loại chức năng để làm việc với các URL (có sẵn từ PHP 4.0.4)	PHP_INI_SYSTEM
user_agent	NULL	Xác định đại diện người sử dụng cho PHP để gửi (có sẵn từ PHP 4.3)	PHP_INI_ALL
default_socket_timeout	"60"	Thiết lập thời gian chờ mặc định, trong vài giây, cho dòng ổ cắm dựa trên (có sẵn từ PHP 4.3)	PHP_INI_ALL
từ	""	Xác định mật khẩu FTP ẩn danh (địa chỉ email của bạn)	PHP_INI_ALL
auto_detect_line_endings	"0"	Khi thiết lập để "1", PHP sẽ xem xét các dữ liệu đọc bởi fgets () và tập tin () để xem nếu nó được sử dụng Unix, MS-Dos hoặc Mac dòng ký tự kết thúc (có sẵn từ PHP 4.3)	PHP_INI_ALL

Unix / Windows Khả năng tương thích

Khi xác định một con đường trên các nền tảng Unix, dấu gạch chéo (/) được sử dụng như là dấu phân cách thư mục. Tuy nhiên, trên nền tảng Windows, cả hai dấu gạch chéo (/) và dấu gạch chéo ngược (\) có thể được sử dụng.

PHP hệ thống tập tin Chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	PHP
<u>basename ()</u>	Trả về thành phần tên tập tin của một con đường	3
<u>chgrp ()</u>	Thay đổi các nhóm tập tin	3
<u>chmod ()</u>	Thay đổi chế độ tập tin	3
<u>chown ()</u>	Thay đổi chủ sở hữu tập tin	3

<u>clearstatcache ()</u>	Xóa bộ nhớ cache tình trạng tập tin	3
<u>copy ()</u>	Sao chép một tập tin	3
<u>delete ()</u>	Xem bỏ liên kết () hoặc unset ()	
<u>dirname ()</u>	Trả về thành phần tên thư mục của một con đường	3
<u>disk_free_space ()</u>	Trả về không gian miễn phí của một thư mục	4
<u>disk_total_space ()</u>	Trả về tổng kích thước của một thư mục	4
<u>diskfreespace ()</u>	Bí danh của disk_free_space ()	3
<u>fclose ()</u>	Đóng một tập tin mở	3
<u>feof ()</u>	Các thử nghiệm về cuối tập tin vào một tập tin mở	3
<u>fflush ()</u>	Flushes đệm đầu ra vào một tập tin mở	4
<u>fgetc ()</u>	Trả về một nhân vật từ một tập tin mở	3
<u>fgetcsv ()</u>	Phân tích một dòng từ một tập tin mở, kiểm tra các lĩnh vực CSV	3
<u>fgets ()</u>	Trả về một dòng từ một tập tin mở	3
<u>fgetss ()</u>	Trả về một dòng, với các thẻ HTML và PHP loại bỏ, từ một tập tin mở	3
<u>file ()</u>	Đọc một tập tin vào một mảng	3
<u>file_exists ()</u>	Kiểm tra có hay không một tập tin hoặc thư mục tồn tại	3
<u>file_get_contents ()</u>	Đọc một tập tin vào một chuỗi	4
<u>file_put_contents</u>	Ghi một chuỗi vào một tập tin	5
<u>fileatime ()</u>	Trả về thời gian truy cập cuối cùng của một tập tin	3
<u>filectime ()</u>	Trả về thời gian thay đổi cuối cùng của một tập tin	3
<u>filegroup ()</u>	Trả về ID nhóm của một tập tin	3
<u>fileinode ()</u>	Trả về số inode của một tập tin	3
<u>filemtime ()</u>	Trả về thời gian sửa đổi cuối cùng của một tập tin	3
<u>fileowner ()</u>	Trả về ID người dùng (chủ sở hữu) của một tập tin	3
<u>fileperms ()</u>	Trả về quyền truy cập của một tập tin	3
<u>filesize ()</u>	Trả về kích thước tập tin	3
<u>filetype ()</u>	Trả về các loại tập tin	3

<u>flock ()</u>	Khóa hoặc phát hành một tập tin	3
<u>fnmatch ()</u>	Kết quả các trận đấu một tên tập tin hoặc chuỗi với một mô hình cụ thể	4
<u>fopen ()</u>	Mở một tập tin hoặc URL	3
<u>fpassthru ()</u>	Đọc từ một tập tin mở, cho đến khi kết thúc tập tin, và ghi kết quả vào bộ đệm đầu ra	3
<u>fputcsv ()</u>	Định dạng một dòng như CSV và viết nó vào một tập tin mở	5
<u>fputs ()</u>	Bí danh của fwrite ()	3
<u>fread ()</u>	Đọc từ một tập tin mở	3
<u>fscanf ()</u>	Phân tích đầu vào từ một tập tin mở theo một định dạng được chỉ định	4
<u>fseek ()</u>	Tìm kiếm trong một tập tin mở	3
<u>fstat ()</u>	Trả về thông tin về một tập tin mở	4
<u>ftell ()</u>	Trả về vị trí hiện tại trong một tập tin mở	3
<u>ftruncate ()</u>	Cắt bớt một tập tin mở với một chiều dài quy định	4
<u>fwrite ()</u>	Viết cho một tập tin mở	3
<u>glob ()</u>	Trả về một mảng các tên tập tin / thư mục phù hợp với một mẫu quy định	4
<u>is_dir ()</u>	Kiểm tra xem một tập tin là một thư mục	3
<u>is_executable ()</u>	Kiểm tra xem một tập tin thực thi	3
<u>is_file ()</u>	Kiểm tra xem một tập tin là một tập tin thường xuyên	3
<u>is_link ()</u>	Kiểm tra xem một tập tin là một liên kết	3
<u>is_readable ()</u>	Kiểm tra xem một tập tin có thể đọc được	3
<u>is_uploaded_file ()</u>	Kiểm tra xem một tập tin đã được tải lên qua HTTP POST	3
<u>is_writable ()</u>	Kiểm tra xem một tập tin là ghi được	4
<u>is_writeable ()</u>	Bí danh của is_writable ()	3
<u>link ()</u>	Tạo một liên kết cứng	3
<u>linkinfo ()</u>	Trả về thông tin về một liên kết cứng	3
<u>lstat ()</u>	Trả về thông tin về một tập tin hoặc liên kết tượng trưng	3
<u>mkdir ()</u>	Tạo một thư mục	3
<u>move_uploaded_file ()</u>	Di chuyển một tập tin được tải lên một vị trí mới	4

<u>parse_ini_file ()</u>	Phân tích một tập tin cấu hình	4
<u>pathinfo ()</u>	Trả về thông tin về một đường dẫn tập tin	4
<u>pclose ()</u>	Đóng cửa một đường ống được mở bởi popen ()	3
<u>popen ()</u>	Mở một đường ống	3
<u>readfile ()</u>	Đọc một tập tin và viết nó vào bộ đệm đầu ra	3
<u>readlink ()</u>	Trả về mục tiêu của một liên kết mềm	3
<u>realpath ()</u>	Trả về tên đường dẫn tuyệt đối	4
<u>rename ()</u>	Đổi tên một tập tin hoặc thư mục	3
<u>rewind ()</u>	Tua lại một con trỏ tập tin	3
<u>rmdir ()</u>	Loại bỏ một thư mục rỗng	3
<u>set_file_buffer ()</u>	Thiết lập kích thước bộ đệm của một tập tin mở	3
<u>stat ()</u>	Trả về thông tin về một tập tin	3
<u>symlink ()</u>	Tạo một liên kết tượng trưng	3
<u>tempnam ()</u>	Tạo ra một tập tin duy nhất tạm thời	3
<u>tmpfile ()</u>	Tạo ra một tập tin duy nhất tạm thời	3
<u>touch ()</u>	Thiết lập thời gian truy cập và sửa đổi của một tập tin	3
<u>umask ()</u>	Thay đổi quyền truy cập tập tin cho các tập tin	3
<u>unlink ()</u>	Xóa một tập tin	3

PHP hệ thống tập tin hằng số

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ các hằng số.

Liên tục	Mô tả	P
GLOB_BRACE		
GLOB_ONLYDIR		
GLOB_MARK		
GLOB_NOSORT		
GLOB_NOCHECK		

GLOB_NOESCAPE	
PATHINFO_DIRNAME	
PATHINFO_BASENAME	
PATHINFO_EXTENSION	
FILE_USE_INCLUDE_PATH	
FILE_APPEND	
FILE_IGNORE_NEW_LINES	
FILE_SKIP_EMPTY_LINES	

PHP lọc Chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
<u>filter_has_var ()</u>	Kiểm tra nếu một biến của một loại đầu vào quy định tồn tại	5
<u>filter_id ()</u>	Trả về số ID của một bộ lọc xác định	5
<u>filter_input ()</u>	Có được đầu vào từ bên ngoài kịch bản và lọc nó	5
<u>filter_input_array ()</u>	Nhận nhiều đầu vào từ bên ngoài kịch bản và lọc chúng	5
<u>filter_list ()</u>	Trả về một mảng của tất cả các bộ lọc được hỗ trợ	5
<u>filter_var_array ()</u>	Nhận nhiều biến số và lọc chúng	5
<u>filter_var ()</u>	Một biến và lọc nó	5

PHP bộ lọc

ID Name	Mô tả
<u>FILTER_CALLBACK</u>	Gọi một chức năng người dùng định nghĩa để lọc dữ liệu
<u>FILTER_SANITIZE_STRING</u>	Thẻ Strip, tùy chọn tước bỏ hoặc mã hóa các ký tự đặc biệt
<u>FILTER_SANITIZE_STRIPPED</u>	Bí danh của bộ lọc "chuỗi"
<u>FILTER_SANITIZE_ENCODED</u>	URL mã hóa chuỗi, tùy chọn tước bỏ hoặc mã hóa các ký tự đặc biệt

<u><code>FILTER_SANITIZE_SPECIAL_CHARS</code></u>	HTML-escape "<> &" và các ký tự có giá trị ASCII ít hơn 32
<u><code>FILTER_SANITIZE_EMAIL</code></u>	Hủy bỏ tất cả các ký tự, ngoại trừ chữ cái, chữ số và <code>\$%&'*+ - / = ^ _ ` ? . { } ~ @ []</code>
<u><code>FILTER_SANITIZE_URL</code></u>	Hủy bỏ tất cả các ký tự, ngoại trừ chữ cái, chữ số và <code>\$+ * ' () , { } \ \ ^ ~ [] ` < > # % " ; : @ & =</code>
<u><code>FILTER_SANITIZE_NUMBER_INT</code></u>	Hủy bỏ tất cả các ký tự, ngoại trừ chữ số và <code>+ -</code>
<u><code>FILTER_SANITIZE_NUMBER_FLOAT</code></u>	Hủy bỏ tất cả các ký tự, ngoại trừ chữ số, <code>+ -</code> , và tùy chọn, <code>eE</code> .
<u><code>FILTER_SANITIZE_MAGIC_QUOTES</code></u>	Áp dụng addslashes ()
<u><code>FILTER_UNSAFE_RAW</code></u>	Không làm gì cả, tùy chọn tước bỏ hoặc mã hóa các ký tự đặc biệt
<u><code>FILTER_VALIDATE_INT</code></u>	Xác nhận giá trị là số nguyên, tùy chọn từ phạm vi chỉ định
<u><code>FILTER_VALIDATE_BOOLEAN</code></u>	Trở về TRUE "1", "true", "trên" và "có", FALSE cho "0", "false", "tắt", "không", và "", NULL nếu không
<u><code>FILTER_VALIDATE_FLOAT</code></u>	Xác nhận giá trị như phao
<u><code>FILTER_VALIDATE_REGEXP</code></u>	Xác nhận giá trị so với regexp, một biểu thức chính quy tương thích với Perl-
<u><code>FILTER_VALIDATE_URL</code></u>	Xác nhận giá trị như URL, tùy chọn với các thành phần cần thiết
<u><code>FILTER_VALIDATE_EMAIL</code></u>	Xác nhận giá trị như e-mail
<u><code>FILTER_VALIDATE_IP</code></u>	Xác nhận giá trị như địa chỉ IP, tùy chọn chỉ IPv4 hoặc IPv6 hay không phải từ phạm vi tư nhân hoặc dành riêng

PHP FTP Giới thiệu

Các chức năng FTP cung cấp cho khách hàng truy cập vào tập tin máy chủ thông qua File Transfer Protocol (FTP).

Các chức năng FTP được sử dụng để mở, đăng nhập và kết nối chặt chẽ, cũng như tải lên, tải về, đổi tên, xóa, và nhận được thông tin về các tập tin từ các máy chủ tập tin. Không phải tất cả các chức năng FTP sẽ làm việc với tất cả các máy chủ hoặc trả lại kết quả tương tự. Các chức năng FTP trở thành có sẵn với PHP 3.

Các chức năng này có nghĩa là để truy cập chi tiết đến một máy chủ FTP. Nếu bạn chỉ muốn đọc từ hay ghi vào một tập tin trên một máy chủ FTP, xem xét việc sử dụng wrapper ftp:// với các chức năng hệ thống tập tin.

Cài đặt

Phiên bản cửa sổ của PHP đã được xây dựng trong hỗ trợ cho các phần mở rộng FTP. Vì vậy, các chức năng FTP sẽ làm việc tự động.

Tuy nhiên, nếu bạn đang chạy phiên bản Linux của PHP, bạn sẽ có để biên dịch PHP với - *enable-ftp* (PHP 4 +) hoặc- *with-ftp* (PHP 3) để có được các chức năng FTP để làm việc.

PHP FTP Chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
<u>ftp_alloc ()</u>	Phân bổ không gian cho một tập tin được tải lên máy chủ FTP	5
<u>ftp_cdup ()</u>	Thay đổi thư mục hiện hành đến thư mục cha trên máy chủ FTP	3
<u>ftp_chdir ()</u>	Thay đổi thư mục hiện hành trên máy chủ FTP	3
<u>ftp_chmod ()</u>	Bộ quyền truy cập vào một tập tin thông qua FTP	5
<u>ftp_close ()</u>	Đóng một kết nối FTP	4
<u>ftp_connect ()</u>	Mở một kết nối FTP	3
<u>ftp_delete ()</u>	Xóa một tập tin trên máy chủ FTP	3
<u>ftp_exec ()</u>	Thực thi một chương trình / lệnh trên máy chủ FTP	4
<u>ftp_fget ()</u>	Tải về một tập tin từ máy chủ FTP và lưu nó vào một tập tin mở	3
<u>ftp_fput ()</u>	Cập nhật từ một tập tin mở và lưu nó vào một tập tin trên máy chủ FTP	3
<u>ftp_get_option ()</u>	Trả về hành vi thời gian chạy của các kết nối FTP	4
<u>ftp_get ()</u>	Tải một tập tin từ máy chủ FTP	3
<u>ftp_login ()</u>	Bản ghi trên một kết nối FTP	3
<u>ftp_mdtm ()</u>	Trả về thời gian sửa đổi cuối cùng của một tập tin được chỉ định	3
<u>ftp_mkdir ()</u>	Tạo một thư mục mới trên máy chủ FTP	3
<u>ftp_nb_continue ()</u>	Tiếp tục lấy / gửi một tập tin (non-blocking)	4
<u>ftp_nb_fget ()</u>	Tải một tập tin từ máy chủ FTP và lưu nó vào một tập tin mở (non-blocking)	4

<u>ftp_nb_fput ()</u>	Cập nhật từ một tập tin mở và lưu nó vào một tập tin trên máy chủ FTP (non-blocking)	4
<u>ftp_nb_get ()</u>	Tải một tập tin từ máy chủ FTP (non-blocking)	4
<u>ftp_nb_put ()</u>	Tải lên một tập tin đến máy chủ FTP (non-blocking)	4
<u>ftp_nlist ()</u>	Danh sách các tập tin trong một thư mục được chỉ định trên máy chủ FTP	3
<u>ftp_pasv ()</u>	Bật chế độ thụ động hoặc tắt	3
<u>ftp_put ()</u>	Tải lên một tập tin đến máy chủ FTP	3
<u>ftp_pwd ()</u>	Trả về tên thư mục hiện hành	3
<u>ftp_quit ()</u>	Bí danh của ftp_close ()	3
<u>ftp_raw ()</u>	Gửi một lệnh liệu đến máy chủ FTP	5
<u>ftp_rawlist ()</u>	Trả về một danh sách chi tiết của các tập tin trong thư mục chỉ định	3
<u>ftp_rename ()</u>	Đặt lại tên một tập tin hoặc thư mục trên máy chủ FTP	3
<u>ftp_rmdir ()</u>	Loại bỏ một thư mục trên máy chủ FTP	3
<u>ftp_set_option ()</u>	Thiết lập các tùy chọn thời gian chạy cho kết nối FTP	4
<u>ftp_site ()</u>	Gửi một lệnh SITE đến máy chủ	3
<u>ftp_size ()</u>	Trả về kích thước của các tập tin được chỉ định	3
<u>ftp_ssl_connect ()</u>	Mở một kết nối an toàn SSL FTP	4
<u>ftp_systype ()</u>	Trả về định danh kiểu hệ thống của máy chủ FTP	3

PHP FTP hằng số

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ các hằng số.

Liên tục	Mô tả	P
FTP_ASCII		3
FTP_TEXT		3
FTP_BINARY		3
FTP_IMAGE		3
FTP_TIMEOUT_SEC		3

FTP_AUTOSEEK		4
FTP_AUTORESUME	Xác định tiếp tục vị trí vị trí và bắt đầu nhận và đặt yêu cầu tự động	4
FTP_FAILED	Không đồng bộ chuyển giao đã thất bại	4
FTP_FINISHED	Không đồng bộ chuyển giao đã hoàn thành	4
FTP_MOREDATA	Vẫn còn hoạt động không đồng bộ chuyển giao	4

PHP HTTP Giới thiệu

Các chức năng HTTP cho phép bạn thao tác thông tin được gửi đến trình duyệt bởi các máy chủ Web, trước khi bất kỳ sản lượng khác đã được gửi.

Cài đặt

Các chức năng thư mục là một phần của lõi PHP. Không có cài đặt cần thiết để sử dụng các chức năng này.

PHP HTTP Chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
<u>header ()</u>	Sẽ gửi một header HTTP liệu cho một khách hàng	3
<u>headers_list ()</u>	Trả về một danh sách các tiêu đề phản ứng gửi (hoặc sẵn sàng để gửi)	5
<u>headers_sent ()</u>	Kiểm tra nếu / nơi các tiêu đề HTTP đã được gửi	3
<u>setcookie ()</u>	Gửi một cookie HTTP với một khách hàng	3
<u>setrawcookie ()</u>	Gửi một cookie HTTP mà không có URL mã hóa các giá trị cookie	5

Giới thiệu PHP libxml

Các chức năng và hằng số libxml được sử dụng cùng với các chức năng SimpleXML, XSLT và DOM.

Cài đặt

Các chức năng này yêu cầu các gói libxml [Tải về tại xmlsoft.org](http://xmlsoft.org)

PHP libxml Chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
libxml_clear_errors ()	Rõ ràng libxml lỗi bộ đệm	5
libxml_get_errors ()	Lấy mảng lỗi	5
libxml_get_last_error ()	Lấy lỗi cuối cùng từ libxml	5
libxml_set_streams_context ()	Đặt bối cảnh suối để tải tài liệu libxml tiếp theo hoặc viết	5
libxml_use_internal_errors ()	Vô hiệu hoá lỗi libxml và cho phép người sử dụng để lấy thông tin lỗi khi cần thiết	5

PHP libxml hằng số

Chức năng	Mô tả	P
LIBXML_COMPACT	Thiết lập tối ưu hóa phân bổ các nút nhỏ. Điều này có thể cải thiện hiệu suất ứng dụng	5
LIBXML_DTDATTR	Thiết lập mặc định DTD thuộc tính	5
LIBXML_DTDLOAD	Tải tập hợp con bên ngoài	5
LIBXML_DTDVALID	Xác nhận với các DTD	5
LIBXML_NOBLANKS	Loại bỏ các nút trống	5
LIBXML_NOCDATA	Đặt CDATA như các nút văn bản	5
LIBXML_NOEMPTYTAG	Thay đổi thẻ rỗng (ví dụ như </ br>), chỉ có sẵn trong lưu DOMDocument-> () và DOMDocument-> saveXML () chức năng	5
LIBXML_NOENT	Thay thế các thực thể	5
LIBXML_NOERROR	Không hiển thị các báo cáo lỗi	5
LIBXML_NONET	Ngăn chặn truy cập mạng trong khi tải tài liệu	5

LIBXML_NOWARNING	Không hiển thị báo cáo cảnh báo	5
LIBXML_NOXMLDECL	Thả khai báo XML khi tiết kiệm một tài liệu	5
LIBXML_NSCLEAN	Hủy bỏ các tờ khai không gian tên dư thừa	5
LIBXML_XINCLUDE	Sử dụng XInclude thay thế	5
LIBXML_ERR_ERROR	Nhận được lỗi thu hồi	5
LIBXML_ERR_FATAL	Nhận lỗi nghiêm trọng	5
LIBXML_ERR_NONE	Không có lỗi	5
LIBXML_ERR_WARNING	Nhận được cảnh báo đơn giản	5
LIBXML_VERSION	Nhận libxml phiên bản (ví dụ như 20.605 hoặc 20.617)	5
LIBXML_DOTTED_VERSION	Chấm libxml phiên bản (ví dụ như 2.6.5 hoặc 2.6.17)	5

PHP Thư Giới thiệu

Hàm mail () cho phép bạn gửi email trực tiếp từ một kịch bản.

Yêu cầu

Đối với các chức năng mail sẽ có sẵn, PHP đòi hỏi một hệ thống email được cài đặt và hoạt động. Chương trình được sử dụng được xác định bởi các thiết lập cấu hình trong file php.ini.

Cài đặt

Các chức năng mail này là một phần của lõi PHP. Không có cài đặt cần thiết để sử dụng các chức năng này.

Thời gian chạy cấu hình

Các hành vi của các chức năng thư bị ảnh hưởng bởi các thiết lập trong file php.ini.

Thư cấu hình tùy chọn:

Tên	Mặc định	Mô tả	Thay đổi
-----	----------	-------	----------

SMTP	"localhost"	Windows chỉ: Tên DNS hoặc địa chỉ IP của máy chủ SMTP	PHP_INI_ALL
smtp_port	"25"	Windows chỉ: số cổng SMTP. Có sẵn từ PHP 4.3	PHP_INI_ALL
sendmail_from	NULL	Windows chỉ: Chỉ định địa chỉ "từ" được sử dụng trong email gửi từ PHP	PHP_INI_ALL
sendmail_path	NULL	Các hệ thống Unix: Xác định nơi mà chương trình sendmail có thể được tìm thấy (thường là /usr / sbin / sendmail hoặc /usr / lib / sendmail)	PHP_INI_SYSTEM

PHP Thư Chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
ezmlm_hash ()	Tính giá trị băm cần thiết bởi hệ thống danh sách gửi thư EZMLM	3
<u>mail ()</u>	Cho phép bạn gửi email trực tiếp từ một kịch bản	3

PHP Toán Giới thiệu

Các chức năng toán học có thể xử lý các giá trị trong phạm vi của các số nguyên và các loại phao.

Cài đặt

Các chức năng toán học là một phần của lõi PHP. Không có cài đặt cần thiết để sử dụng các chức năng này.

PHP Math Chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
<u>abs ()</u>	Trả về giá trị tuyệt đối của một số	3
<u>acos ()</u>	Trả về arccosine của một số	3

<u>acosh ()</u>	Trả về cosin hyperbol nghịch đảo của một số	4
<u>asin ()</u>	Trả về arcsine của một số	3
<u>asinh ()</u>	Trả về sin hyperbol nghịch đảo của một số	4
<u>atan ()</u>	Trả về arctangent của một số như là một giá trị số giữa-PI / 2 và PI / 2 radian	3
<u>atan2 ()</u>	Trả về theta góc của một điểm (x, y) như là một giá trị số giữa-PI và PI radian	3
<u>atanh ()</u>	Trả về tang hyperbol nghịch đảo của một số	4
<u>base_convert ()</u>	Chuyển đổi một số từ một cơ sở khác	3
<u>bindec ()</u>	Chuyển đổi một số nhị phân cho một số thập phân	3
<u>ceil ()</u>	Trả về giá trị của một số được làm tròn lên đến số nguyên gần nhất	3
<u>cos ()</u>	Trả về cosin của một số	3
<u>cosh ()</u>	Trả về cosin hyperbol của một số	4
<u>decbin ()</u>	Chuyển đổi một số thập phân một số nhị phân	3
<u>dechex ()</u>	Chuyển đổi một số thập phân cho một số thập lục phân	3
<u>decoct ()</u>	Chuyển đổi một số thập phân một số bát phân	3
<u>deg2rad ()</u>	Chuyển đổi một mức độ một số radian	3
<u>exp ()</u>	Trả về giá trị của E^x	3
<u>expm1 ()</u>	Trả về giá trị của $e^x - 1$	4
<u>floor ()</u>	Trả về giá trị của một số được làm tròn xuống số nguyên gần nhất	3
<u>fmod ()</u>	Trả về còn lại (modulo) của sự phân chia của các đối số	4
<u>getrandmax ()</u>	Trả về số lượng tối đa ngẫu nhiên mà có thể được trả lại bởi một cuộc gọi đến các rand () chức năng	3
<u>hexdec ()</u>	Chuyển đổi một số thập lục phân cho một số thập phân	3
<u>hypot ()</u>	Trả về độ dài của cạnh huyền của một tam giác vuông góc	4
<u>is_finite ()</u>	Trả về true nếu giá trị là một số hữu hạn	4
<u>is_infinite ()</u>	Trả về true nếu giá trị là một số lượng vô hạn	4
<u>is_nan ()</u>	Trả về true nếu một giá trị không phải là số	4
<u>lcg_value ()</u>	Trả về một số ngẫu nhiên trong khoảng (0,1)	4

<u>log ()</u>	Trả về logarit tự nhiên (cơ sở E) của một số	3
<u>log10 ()</u>	Trả về logarit cơ sở-10 của một số	3
<u>log1p ()</u>	Trả về log (1 + số)	4
<u>max ()</u>	Trả về số lượng với giá trị cao nhất của hai số đã xác định	3
<u>min ()</u>	Trả về số lượng với giá trị thấp nhất của hai số đã xác định	3
<u>mt_getrandmax ()</u>	Trả về giá trị lớn nhất có thể có thể được trả lại bởi mt_rand ()	3
<u>mt_rand ()</u>	Trả về một số nguyên ngẫu nhiên bằng cách sử dụng thuật toán Mersenne Twister	3
<u>mt_srand ()</u>	Hạt giống Mersenne Twister ngẫu nhiên số máy phát điện	3
<u>octdec ()</u>	Chuyển đổi một số bát phân cho một số thập phân	3
<u>pi ()</u>	Trả về giá trị của PI	3
<u>pow ()</u>	Trả về giá trị của x để sức mạnh của y	3
<u>rad2deg ()</u>	Chuyển đổi một số radian đến một mức độ	3
<u>rand ()</u>	Trả về một số nguyên ngẫu nhiên	3
<u>round ()</u>	Tròn một số đến số nguyên gần nhất	3
<u>sin ()</u>	Trả về sin của một số	3
<u>sinh ()</u>	Trả về sin hyperbol của một số	4
<u>sqrt ()</u>	Trả về căn bậc hai của một số	3
<u>srand ()</u>	Hạt giống máy phát điện số ngẫu nhiên	3
<u>tan ()</u>	Trả về tang của một góc	3
<u>tanh ()</u>	Trả về tang hyperbol của một góc	4

PHP Math hằng số

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ các hằng số.

Liên tục	Mô tả	P
M_E	Trả về e (khoảng 2,718)	4
M_EULER	Trả về hằng số Euler (khoảng 0,577)	4

M_LNPI	Trả về logarit tự nhiên của PI (khoảng 1,144)	4
M_LN2	Trả về logarit tự nhiên của 2 (khoảng 0,693)	4
M_LN10	Trả về logarit tự nhiên của 10 (khoảng 2,302)	4
M_LOG2E	Trả về logarit cơ sở-2 của E (khoảng 1,442)	4
M_LOG10E	Trả về logarit cơ sở-10 E (khoảng 0,434)	4
M_PI	Trả về PI (khoảng 3,14159)	3
M_PI_2	Trả về PI / 2 (khoảng 1,570)	4
M_PI_4	Trả về PI / 4 (khoảng 0,785)	4
M_1_PI	Trả về 1/PI (khoảng 0,318)	4
M_2_PI	Trả về 2/PI (khoảng 0,636)	4
M_SQRTPI	Trả về căn bậc hai của PI (khoảng 1,772)	4
M_2_SQRTPI	Trả về 2/square gốc của PI (khoảng 1,128)	4
M_SQRT1_2	Trả về căn bậc hai của 1/2 (khoảng 0,707)	4
M_SQRT2	Trả về căn bậc hai của 2 (khoảng 1,414)	4
M_SQRT3	Trả về căn bậc hai của 3 (khoảng 1,732)	4

Giới thiệu PHP Miscellaneous

Misc. chức năng chỉ được đặt ở đây bởi vì không ai trong số các loại khác dường như để phù hợp với.

Cài đặt

Các chức năng misc là một phần của lõi PHP. Không có cài đặt cần thiết để sử dụng các chức năng này.

Thời gian chạy cấu hình

Các hành vi của các chức năng misc bị ảnh hưởng bởi các thiết lập trong file php.ini.

Misc. tùy chọn cấu hình:

Tên	Mặc định	Mô tả	Thay đổi
ignore_user_abort	"0"	FALSE chỉ ra rằng các kịch bản sẽ được chấm dứt ngay sau khi họ cố gắng để sản xuất một cái gì đó sau khi khách hàng đã hủy bỏ kết nối của họ	PHP_INI_ALL
highlight.string	"# DD0000"	Màu sắc để làm nổi bật một chuỗi trong PHP cú pháp	PHP_INI_ALL
highlight.comment	"# FF8000"	Màu sắc để làm nổi bật ý kiến PHP	PHP_INI_ALL
highlight.keyword	"007.700"	Màu sắc cho các từ khóa nổi bật cú pháp PHP (ví dụ như dấu ngoặc đơn và dấu chấm phẩy)	PHP_INI_ALL
highlight.bg	"# FFFFFF"	Màu sắc cho nền	PHP_INI_ALL
highlight.default	"# 0000BB"	Mặc định màu sắc cho PHP cú pháp	PHP_INI_ALL
highlight.html	"# 000000"	Màu sắc cho HTML code	PHP_INI_ALL
browscap	NULL	Tên và vị trí của trình duyệt khả năng tập tin (ví dụ như browscap.ini)	PHP_INI_SYSTEM

PHP Misc. Chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
<u>connection_aborted ()</u>	Kiểm tra xem khách hàng đã bị ngắt kết nối	3
<u>connection_status ()</u>	Trả về tình trạng kết nối hiện tại	3
connection_timeout ()	Phản đối trong PHP 4.0.5	3
<u>constant ()</u>	Trả về giá trị của một hằng số	4
<u>define ()</u>	Định nghĩa một hằng số	3
<u>defined ()</u>	Kiểm tra xem một hằng số tồn tại	3
<u>die ()</u>	In một tin nhắn và thoát khỏi kịch bản hiện tại	3
<u>eval ()</u>	Đánh giá một chuỗi như là mã PHP	3
<u>exit ()</u>	In một tin nhắn và thoát khỏi kịch bản hiện tại	3
<u>get_browser ()</u>	Trả về khả năng của trình duyệt của người dùng	3
<u>highlight_file ()</u>	Kết quả đầu ra một tập tin với cú pháp PHP nhấn mạnh	4

<u>highlight_string ()</u>	Kết quả đầu ra một chuỗi với cú pháp PHP nhấn mạnh	4
<u>ignore_user_abort ()</u>	Bộ xem một khách hàng từ xa có thể hủy bỏ các hoạt động của một kịch bản	3
<u>pack ()</u>	Gói dữ liệu vào một chuỗi nhị phân	3
php_check_syntax ()	Phản đối trong PHP 5.0.5	5
<u>php_strip_whitespace ()</u>	Trả về mã nguồn của một tập tin với ý kiến, PHP và khoảng trắng xóa	5
<u>show_source ()</u>	Bí danh của highlight_file ()	4
<u>sleep ()</u>	Sự chậm trễ thực thi mã cho một số giây	3
<u>time_nanosleep ()</u>	Sự chậm trễ thực thi mã cho một số giây và nano giây	5
<u>time_sleep_until ()</u>	Sự chậm trễ mã thực thi cho đến khi một thời gian quy định	5
<u>uniqid ()</u>	Tạo ra một ID duy nhất	3
<u>unpack ()</u>	Giải nén dữ liệu từ một chuỗi nhị phân	3
<u>usleep ()</u>	Sự chậm trễ thực thi mã cho một số micro giây	3

PHP Misc. Hằng số

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ các hằng số.

Liên tục	Mô tả	P
CONNECTION_ABORTED		
CONNECTION_NORMAL		
CONNECTION_TIMEOUT		
__COMPILER_HALT_OFFSET__		5

PHP MySQL Giới thiệu

Các chức năng MySQL cho phép bạn truy cập vào các máy chủ cơ sở dữ liệu MySQL.

Cài đặt

Đối với các chức năng MySQL sẽ có sẵn, bạn phải biên dịch PHP với MySQL hỗ trợ.

Để biên dịch, sử dụng - *with-mysql* = *DIR* (*DIR* tùy chọn điểm đến thư mục MySQL).

Lưu ý: Đối với đầy đủ chức năng của MySQL phiên bản lớn hơn trên 4,1, sử dụng phần mở rộng MySQLi thay thế. Nếu bạn muốn cài đặt cả phần mở rộng mysql và mở rộng mysqli bạn nên sử dụng cùng một thư viện khách hàng để tránh bất kỳ xung đột.

Cài đặt trên hệ thống Linux

PHP 5 +: MySQL và thư viện MySQL không được kích hoạt theo mặc định. Sử dụng - *with-mysql* = *DIR* cấu hình tùy chọn để bao gồm MySQL hỗ trợ và tải về tiêu đề và thư viện từ www.mysql.com.

Các hệ thống cài đặt trên

PHP 5 + MySQL không được kích hoạt theo mặc định, vì vậy php_mysql.dll phải được kích hoạt bên trong php.ini. Ngoài ra, PHP cần truy cập vào thư viện khách MySQL. Một tập tin có tên libmysql.dll được bao gồm trong phân phối Windows PHP, và để cho PHP nói chuyện với MySQL tập tin này cần phải có sẵn để PATH hệ thống Windows.

Để kích hoạt bất kỳ phần mở rộng PHP, PHP extension_dir thiết lập (trong file php.ini) nên được đặt vào thư mục nơi mà các phần mở rộng PHP được đặt. Một giá trị extension_dir ví dụ là c: \ php \ ext.

Lưu ý: Nếu bạn nhận được các lỗi sau khi bắt đầu các máy chủ web: "Không thể tải thư viện năng động ' / php_mysql.dll", điều này là bởi vì php_mysql.dll hoặc libmysql.dll không có thể được tìm thấy bởi hệ thống.

Thời gian chạy cấu hình

Các hành vi của các chức năng MySQL bị ảnh hưởng bởi các thiết lập trong file php.ini.

MySQL cấu hình tùy chọn:

Tên	Mặc định	Mô tả	Thay đổi
mysql.allow_persistent	"1"	Có hay không để cho phép các kết nối liên tục	PHP_INI_SYSTEM
mysql.max_persistent	"-1"	Số lượng tối đa các kết nối liên tục cho mỗi quá trình	PHP_INI_SYSTEM
mysql.max_links	"-1"	Số lượng tối đa các kết nối cho mỗi quá trình (kết nối liên tục bao gồm)	PHP_INI_SYSTEM
mysql.trace_mode	"0"	Theo dõi chế độ. Khi thiết lập để "1", cảnh báo và SQL-lỗi sẽ được hiển thị. Có sẵn từ PHP 4.3	PHP_INI_ALL
mysql.default_port	NULL	Các số cổng TCP mặc định sử dụng	PHP_INI_ALL
mysql.default_socket	NULL	Tên ổ cắm mặc định để sử dụng. Có sẵn từ PHP	PHP_INI_ALL

		4.0.1	
mysql.default_host	NULL	Các máy chủ lưu trữ máy chủ mặc định để sử dụng (không áp dụng trong chế độ an toàn SQL)	PHP_INI_ALL
mysql.default_user	NULL	Tên người sử dụng mặc định để sử dụng (không áp dụng trong chế độ an toàn SQL)	PHP_INI_ALL
mysql.default_password	NULL	Mật khẩu mặc định để sử dụng (không áp dụng trong chế độ an toàn SQL)	PHP_INI_ALL
mysql.connect_timeout	"60"	Kết nối thời gian chờ trong vài giây	PHP_INI_ALL

Loại tài nguyên

Có hai loại tài nguyên được sử dụng trong phần mở rộng MySQL. Người đầu tiên là `link_identifier` cho một kết nối cơ sở dữ liệu, thứ hai là một nguồn tài nguyên mà giữ kết quả của một truy vấn.

Lưu ý: Hầu hết các MySQL chức năng chấp nhận `link_identifier` như là các tham số tùy chọn cuối cùng. Nếu nó không được cung cấp, kết nối mở cuối cùng được sử dụng.

PHP MySQL chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
<u>mysql_affected_rows ()</u>	Trả về số hàng bị ảnh hưởng trong hoạt động MySQL trước	3
<u>mysql_change_user ()</u>	Phản đối. Thay đổi người sử dụng kết nối MySQL hiện tại	3
<u>mysql_client_encoding ()</u>	Trả về tên các bộ ký tự cho các kết nối hiện tại	4
<u>mysql_close ()</u>	Đóng kết nối MySQL không liên tục	3
<u>mysql_connect ()</u>	Mở một kết nối không liên tục MySQL	3
<u>mysql_create_db ()</u>	Phản đối. Tạo ra một cơ sở dữ liệu MySQL mới. Sử dụng <u>mysql_query ()</u> để thay thế	3
<u>mysql_data_seek ()</u>	Di chuyển con trỏ kỷ lục	3
<u>mysql_db_name ()</u>	Trả về một tên cơ sở dữ liệu từ một cuộc gọi đến <u>mysql_list_dbs ()</u>	3
<u>mysql_db_query ()</u>	Phản đối. Gửi một truy vấn MySQL. Sử dụng <u>mysql_select_db ()</u> và	3

	mysql_query () để thay thế	
mysql_drop_db ()	Phản đối. Xóa một cơ sở dữ liệu MySQL. Sử dụng mysql_query () để thay thế	3
mysql_errno ()	Trả về số lỗi của các hoạt động mới nhất của MySQL	3
mysql_error ()	Trả về mô tả lỗi của các hoạt động mới nhất của MySQL	3
mysql_escape_string ()	Phản đối. Escapes một chuỗi để sử dụng trong một mysql_query. Sử dụng mysql_real_escape_string () để thay thế	4
mysql_fetch_array ()	Trả về một hàng từ một recordset như là một mảng kết hợp và / hoặc một mảng số	3
mysql_fetch_assoc ()	Trả về một hàng từ một recordset như là một mảng kết hợp	4
mysql_fetch_field ()	Trả về thông tin cột từ một recordset như một đối tượng	3
mysql_fetch_lengths ()	Trả về độ dài của nội dung của từng lĩnh vực trong một hàng kết quả	3
mysql_fetch_object ()	Trả về một hàng từ một recordset như một đối tượng	3
mysql_fetch_row ()	Trả về một hàng từ một recordset như là một mảng số	3
mysql_field_flags ()	Trả về cờ kết hợp với một lĩnh vực trong một recordset	3
mysql_field_len ()	Trả về độ dài tối đa của một trường trong một recordset	3
mysql_field_name ()	Trả về tên của một trường trong một recordset	3
mysql_field_seek ()	Di chuyển con trỏ kết quả cho một lĩnh vực quy định	3
mysql_field_table ()	Trả về tên của bảng lĩnh vực được chỉ định trong	3
mysql_field_type ()	Trả về kiểu của một trường trong một recordset	3
mysql_free_result ()	Bộ nhớ miễn phí kết quả	3
mysql_get_client_info ()	Trả về thông tin khách hàng MySQL	4
mysql_get_host_info ()	Trả về thông tin máy chủ MySQL	4
mysql_get_proto_info ()	Trả về MySQL giao thức thông tin	4
mysql_get_server_info ()	Trả về thông tin máy chủ MySQL	4
mysql_info ()	Trả về thông tin về các truy vấn cuối cùng	4
mysql_insert_id ()	Trả về ID AUTO_INCREMENT tạo ra từ hoạt động INSERT trước đó	3
mysql_list_dbs ()	Danh sách các cơ sở dữ liệu có sẵn trên một máy chủ MySQL	3
mysql_list_fields ()	Phản đối. Danh sách các lĩnh vực MySQL bảng. Sử dụng mysql_query	3

	() để thay thế	
<u>mysql_list_processes ()</u>	Danh sách MySQL quá trình	4
mysql_list_tables ()	Phản đối. Danh sách các bảng trong một cơ sở dữ liệu MySQL. Sử dụng mysql_query () để thay thế	3
<u>mysql_num_fields ()</u>	Trả về số lĩnh vực trong một recordset	3
<u>mysql_num_rows ()</u>	Trả về số hàng trong một recordset	3
<u>mysql_pconnect ()</u>	Mở một kết nối MySQL dai dẳng	3
<u>mysql_ping ()</u>	Ping một kết nối máy chủ hoặc kết nối lại nếu không có kết nối	4
<u>mysql_query ()</u>	Thực thi một truy vấn trên một cơ sở dữ liệu MySQL	3
<u>mysql_real_escape_string ()</u>	Thoát khỏi một chuỗi để sử dụng trong các câu lệnh SQL	4
<u>mysql_result ()</u>	Trả về giá trị của một trường trong một recordset	3
<u>mysql_select_db ()</u>	Thiết lập cơ sở dữ liệu MySQL hoạt động	3
<u>mysql_stat ()</u>	Trả về tình trạng hệ thống hiện tại của máy chủ MySQL	4
mysql_tablename ()	Phản đối. Trả về tên bảng của trường. Sử dụng mysql_query () để thay thế	3
<u>mysql_thread_id ()</u>	Trả về ID thread hiện hành	4
<u>mysql_unbuffered_query ()</u>	Thực thi một truy vấn trên một cơ sở dữ liệu MySQL (mà không cần lấy / đệm các kết quả)	4

PHP MySQL hằng số

Kể từ PHP 4.3 đã có thể để xác định cờ bổ sung cho các chức năng mysql_connect () và mysql_pconnect ():

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ các hằng số.

Liên tục	Mô tả	P
MYSQL_CLIENT_COMPRESS	Sử dụng giao thức nén	4
MYSQL_CLIENT_IGNORE_SPACE	Cho phép không gian sau khi tên hàm	4
MYSQL_CLIENT_INTERACTIVE	Cho phép tương tác thời gian chờ giây không hoạt động trước khi đóng kết nối	4
MYSQL_CLIENT_SSL	Sử dụng mã hóa SSL (chỉ có sẵn với phiên bản 4 + của	4

	thư viện MySQL client)	
--	------------------------	--

Các `mysql_fetch_array()` chức năng sử dụng một hằng số cho các loại khác nhau của các mảng kết quả. Các hằng số sau đây được định nghĩa:

Liên tục	Mô tả	P
<code>MYSQL_ASSOC</code>	Cột được trả về vào mảng với các fieldname như chỉ số mảng	
<code>MYSQL_BOTH</code>	Cột được trả về vào các mảng có chỉ số một số và fieldname như chỉ số mảng	
<code>MYSQL_NUM</code>	Cột được trả về vào các mảng có chỉ số số (chỉ số bắt đầu từ 0)	

PHP SimpleXML Giới thiệu

Các chức năng SimpleXML cho phép bạn chuyển đổi XML vào một đối tượng.

Đối tượng này có thể được xử lý, giống như bất kỳ đối tượng khác, với selectors sở hữu bình thường và vòng lặp mảng.

Một số các chức năng này đòi hỏi PHP mới nhất xây dựng.

Cài đặt

Các chức năng SimpleXML là một phần của lõi PHP. Không có cài đặt cần thiết để sử dụng các chức năng này.

PHP SimpleXML chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
<code>__construct()</code>	Tạo một đối tượng SimpleXMLElement mới	5
<code>addAttribute()</code>	Thêm một thuộc tính đến các yếu tố SimpleXML	5
<code>addChild()</code>	Thêm một phần tử con các yếu tố SimpleXML	5
<code>asXML()</code>	Nhận được một chuỗi XML từ một yếu tố SimpleXML	5
<code>attributes()</code>	Được các thuộc tính của một yếu tố SimpleXML	5

<u>children ()</u>	Được các con của một nút xác định	5
<u>getDocNamespaces ()</u>	Được các không gian tên của một tài liệu XML	5
<u>getName ()</u>	Nhận được tên của một nguyên tố SimpleXML	5
<u>getNamespace ()</u>	Được các không gian tên từ dữ liệu XML	5
<u>registerXPathNamespace ()</u>	Tạo ra một bối cảnh không gian tên cho các truy vấn XPath tiếp theo	5
<u>simplexml_import_dom ()</u>	Được một đối tượng SimpleXMLElement từ một nút DOM	5
<u>simplexml_load_file ()</u>	Được một đối tượng SimpleXMLElement từ một tài liệu XML	5
<u>simplexml_load_string ()</u>	Được một đối tượng SimpleXMLElement từ một chuỗi XML	5
<u>xpath ()</u>	Chạy một truy vấn XPath trên dữ liệu XML	5

PHP chuỗi Giới thiệu

Chuỗi các chức năng cho phép bạn thao tác chuỗi.

Cài đặt

Các chức năng chuỗi là một phần của lõi PHP. Không có cài đặt cần thiết để sử dụng các chức năng này.

PHP chuỗi Chức năng

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
<u>addslashes ()</u>	Trả về một chuỗi với dấu gạch chéo ngược ở phía trước của các nhân vật được chỉ định	4
<u>addslashes ()</u>	Trả về một chuỗi với dấu gạch chéo ngược ở phía trước của các nhân vật được xác định trước	3
<u>bin2hex ()</u>	Chuyển đổi một chuỗi ký tự ASCII với các giá trị thập lục phân	3
<u>chop ()</u>	Bí danh của rtrim ()	3
<u>chr ()</u>	Trả về một nhân vật từ một giá trị ASCII quy định	3
<u>chunk_split ()</u>	Tách một chuỗi thành một loạt các phần nhỏ hơn	3

<u>convert_cyr_string ()</u>	Chuyển đổi một chuỗi từ một bộ ký tự Cyrillic khác	3
<u>convert_uuencode ()</u>	Giải mã một chuỗi uuencoded	5
<u>convert_uuencode ()</u>	Mã hóa một chuỗi bằng cách sử dụng các thuật toán uuencode	5
<u>count_chars ()</u>	Trả bao nhiêu lần một ký tự ASCII xảy ra trong vòng một chuỗi và trả về thông tin	4
<u>crc32 ()</u>	Tính toán CRC 32-bit cho một chuỗi	4
<u>crypt ()</u>	Chuỗi mã hóa một chiều (băm)	3
<u>echo ()</u>	Kết quả đầu ra chuỗi	3
<u>explode ()</u>	Phá vỡ một chuỗi thành một mảng	3
<u>fprintf ()</u>	Viết một chuỗi định dạng cho một dòng sản lượng quy định	5
<u>get_html_translation_table ()</u>	Trả về bảng dịch được sử dụng bởi htmlspecialchars () và htmlentities ()	4
<u>hebrew ()</u>	Chuyển đổi văn bản tiếng Do Thái, văn bản trực quan	3
<u>hebrevc ()</u>	Chuyển đổi văn bản tiếng Do Thái, văn bản hình ảnh và dòng mới (\ n) vào 	3
<u>html_entity_decode ()</u>	Chuyển đổi các thực thể HTML để ký tự	4
<u>htmlentities ()</u>	Chuyển đổi các ký tự để các thực thể HTML	3
<u>htmlspecialchars_decode ()</u>	Chuyển đổi một số các thực thể HTML được xác định trước với các nhân vật	5
<u>htmlspecialchars ()</u>	Chuyển đổi một số ký tự được xác định trước để các thực thể HTML	3
<u>implode ()</u>	Trả về một chuỗi từ các yếu tố của một mảng	3
<u>join ()</u>	Bí danh của implode ()	3
<u>Levenshtein ()</u>	Trả về khoảng cách Levenshtein giữa hai dây	3
<u>localeconv ()</u>	Trả về miền địa phương số và thông tin định dạng tiền tệ	4
<u>LTRIM ()</u>	Strips khoảng trắng từ phía bên trái của một chuỗi	3
<u>md5 ()</u>	Tính MD5 của một chuỗi	3
<u>md5_file ()</u>	Tính MD5 của một tập tin	4
<u>metaphone ()</u>	Tính phím metaphone của một chuỗi	4
<u>money_format ()</u>	Trả về một chuỗi được định dạng như là một chuỗi tiền tệ	4

<u>nl_langinfo ()</u>	Trả về thông tin địa phương cụ thể	4
<u>nl2br ()</u>	Chèn HTML ngắt dòng ở phía trước của mỗi dòng mới trong một chuỗi	3
<u>number_format ()</u>	Định dạng một số với hàng ngàn nhóm	3
<u>ord ()</u>	Trả về giá trị ASCII của ký tự đầu tiên của một chuỗi	3
<u>parse_str ()</u>	Phân tích một chuỗi truy vấn vào các biến	3
<u>print ()</u>	Kết quả đầu ra một chuỗi	3
<u>printf ()</u>	Đầu ra một chuỗi định dạng	3
<u>quoted_printable_decode ()</u>	Giải mã một chuỗi trích dẫn-in	3
<u>quotemeta ()</u>	Dấu ngoặc kép meta ký tự	3
<u>rtrim ()</u>	Strips khoảng trắng từ phía bên phải của một chuỗi	3
<u>setlocale ()</u>	Thiết lập thông tin ngôn ngữ	3
<u>sha1 ()</u>	Tính SHA-1 hash của một chuỗi	4
<u>sha1_file ()</u>	Tính SHA-1 hash của một tập tin	4
<u>similar_text ()</u>	Tính tương đồng giữa hai chuỗi	3
<u>Soundex ()</u>	Tính phím Soundex của một chuỗi	3
<u>sprintf ()</u>	Ghi một chuỗi định dạng cho một biến	3
<u>sscanf ()</u>	Phân tích đầu vào từ một chuỗi theo một định dạng	4
<u>str_ireplace ()</u>	Thay thế một số ký tự trong một chuỗi (trường hợp-insensitive)	5
<u>str_pad ()</u>	Đệm một chuỗi với độ dài mới	4
<u>str_repeat ()</u>	Lặp lại một chuỗi một số quy định của thời gian	4
<u>str_replace ()</u>	Thay thế một số ký tự trong một chuỗi (trường hợp nhạy cảm)	3
<u>str_rot13 ()</u>	Thực hiện mã hóa ROT13 trên một sợi dây	4
<u>str_shuffle ()</u>	Ngẫu nhiên shuffles tất cả các ký tự trong một chuỗi	4
<u>str_split ()</u>	Chia tách một chuỗi thành một mảng	5
<u>str_word_count ()</u>	Đếm số từ trong một chuỗi	4
<u>strcasecmp ()</u>	So sánh hai chuỗi (trường hợp-insensitive)	3
<u>strchr ()</u>	Tìm sự xuất hiện đầu tiên của một chuỗi bên trong một chuỗi	3

	khác (bí danh của strstr ())	
<u>strcmp ()</u>	So sánh hai chuỗi (trường hợp nhạy cảm)	3
<u>strcoll ()</u>	Locale dựa trên chuỗi so sánh	4
<u>strcspn ()</u>	Trả về số của các nhân vật được tìm thấy trong một chuỗi trước khi bất kỳ một phần của một số ký tự được chỉ định được tìm thấy	3
<u>strip_tags ()</u>	Strips thẻ HTML và PHP từ một chuỗi	3
<u>stripslashes ()</u>	Unquotes một chuỗi trích dẫn với addslashes ()	4
<u>stripslashes ()</u>	Unquotes một chuỗi trích dẫn với addslashes ()	3
<u>stripos ()</u>	Trả về vị trí xuất hiện đầu tiên của một chuỗi bên trong một chuỗi khác (trường hợp-insensitive)	5
<u>strstr ()</u>	Tìm sự xuất hiện đầu tiên của một chuỗi bên trong một chuỗi khác (trường hợp-insensitive)	3
<u>strlen ()</u>	Trả về độ dài của một chuỗi	3
<u>strnatcasecmp ()</u>	So sánh hai chuỗi bằng cách sử dụng một thuật toán "trật tự tự nhiên" (trường hợp-insensitive)	4
<u>strnatcmp ()</u>	So sánh hai chuỗi bằng cách sử dụng một thuật toán "trật tự tự nhiên" (trường hợp nhạy cảm)	4
<u>strncasecmp ()</u>	Chuỗi so sánh n ký tự đầu tiên (trường hợp-insensitive)	4
<u>strncmp ()</u>	Chuỗi so sánh n ký tự đầu tiên (trường hợp nhạy cảm)	4
<u>strpbrk ()</u>	Tìm kiếm một chuỗi cho bất kỳ một tập hợp các ký tự	5
<u>strpos ()</u>	Trả về vị trí xuất hiện đầu tiên của một chuỗi bên trong một chuỗi khác (trường hợp nhạy cảm)	3
<u>strrchr ()</u>	Tìm sự xuất hiện cuối cùng của một chuỗi bên trong một chuỗi khác	3
<u>strrev ()</u>	Đảo ngược một chuỗi	3
<u>strripos ()</u>	Tìm vị trí của sự xuất hiện cuối cùng của một chuỗi bên trong một chuỗi khác (trường hợp-insensitive)	5
<u>strrpos ()</u>	Tìm vị trí của sự xuất hiện cuối cùng của một chuỗi bên trong một chuỗi khác (trường hợp nhạy cảm)	3
<u>strspn ()</u>	Trả về số của các nhân vật được tìm thấy trong một chuỗi chỉ chứa các ký tự từ một charlist quy định	3
<u>strstr ()</u>	Tìm sự xuất hiện đầu tiên của một chuỗi bên trong một chuỗi khác (trường hợp nhạy cảm)	3

<u>strtok ()</u>	Chia tách một chuỗi thành chuỗi nhỏ hơn	3
<u>strtolower ()</u>	Chuyển đổi một chuỗi thành chữ thường chữ	3
<u>strtoupper ()</u>	Chuyển đổi một chuỗi chữ hoa chữ	3
<u>strtr ()</u>	Dịch một số ký tự trong một chuỗi	3
<u>substr ()</u>	Trả về một phần của một chuỗi	3
<u>substr_compare ()</u>	So sánh hai chuỗi từ một vị trí bắt đầu quy định (nhị phân an toàn, và tùy chọn trường hợp nhạy cảm)	5
<u>substr_count ()</u>	Đếm số lần xảy ra một xâu con trong một chuỗi	4
<u>substr_replace ()</u>	Thay thế một phần của một chuỗi với chuỗi khác	4
<u>trim ()</u>	Strips khoảng trắng từ cả hai mặt của một chuỗi	3
<u>ucfirst ()</u>	Chuyển đổi ký tự đầu tiên của một chuỗi thành chữ hoa	3
<u>ucwords ()</u>	Chuyển đổi ký tự đầu tiên của mỗi từ trong một chuỗi thành chữ hoa	3
<u>vfprintf ()</u>	Viết một chuỗi định dạng cho một dòng sản lượng quy định	5
<u>vprintf ()</u>	Đầu ra một chuỗi định dạng	4
<u>vsprintf ()</u>	Ghi một chuỗi định dạng cho một biến	4
<u>WordWrap ()</u>	Kết thúc tốt đẹp một chuỗi đến một số lượng nhất định các ký tự	4

PHP chuỗi hằng số

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ các hằng số.

Liên tục	Mô tả	P
CRIPT_SALT_LENGTH	Chứa chiều dài của phương pháp mã hóa mặc định cho hệ thống. Để mã hóa tiêu chuẩn DES, chiều dài là 2	
CRIPT_STD_DES	Thiết lập để 1 nếu tiêu chuẩn mã hóa DES-với một muối ký tự 2 được hỗ trợ, 0 nếu ngược lại	
CRIPT_EXT_DES	Thiết lập để 1 nếu DES-mã hóa mở rộng dựa trên với một muối ký tự 9 được hỗ trợ, 0 nếu ngược lại	
CRIPT_MD5	Thiết lập để 1 nếu mã hóa MD5 với muối ký tự 12 bắt đầu với \$ 1 được hỗ trợ, 0 nếu ngược lại	

CRYPT_BLOWFISH	Thiết lập để 1 nếu mã hóa Blowfish với muối ký tự 16 bắt đầu với \$ 2 \$ hoặc \$ 2a \$ được hỗ trợ, 0 otherwise0
HTML_SPECIALCHARS	
HTML_ENTITIES	
ENT_COMPAT	
ENT_QUOTES	
ENT_NOQUOTES	
CHAR_MAX	
LC_CTYPE	
LC_NUMERIC	
LC_TIME	
LC_COLLATE	
LC_MONETARY	
LC_ALL	
LC_MESSAGES	
STR_PAD_LEFT	
STR_PAD_RIGHT	
STR_PAD_BOTH	

PHP XML Parser Giới thiệu

Các chức năng XML cho phép bạn phân tích, nhưng không xác nhận, các tài liệu XML.

XML là một định dạng dữ liệu để trao đổi tài liệu có cấu trúc được tiêu chuẩn hóa. Thông tin thêm về XML có thể được tìm thấy trong [Hướng dẫn](#) của chúng tôi [XML](#) .

Phần mở rộng này sử dụng phân tích cú pháp XML Expat.

Người nước ngoài là một bộ phân tích dựa trên sự kiện, xem một tài liệu XML như là một loạt các sự kiện. Khi một sự kiện xảy ra, nó gọi một chức năng cụ thể để xử lý nó.

Người nước ngoài là một phân tích cú pháp không kiểm duyệt, và bỏ qua bất kỳ các DTD được liên kết đến một tài liệu. Tuy nhiên, nếu tài liệu được không cũng được hình thành, nó sẽ kết thúc với một thông báo lỗi.

Bởi vì nó là dựa trên sự kiện, phân tích cú pháp xác nhận không, Người nước ngoài là nhanh chóng và thích hợp cho các ứng dụng web.

Các chức năng phân tích cú pháp XML cho phép bạn tạo ra các phân tích cú pháp XML và xác định xử lý cho các sự kiện XML.

Cài đặt

Các chức năng XML là một phần của lõi PHP. Không có cài đặt cần thiết để sử dụng các chức năng này.

Chức năng phân tích cú pháp XML PHP

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
utf8_decode()	Giải mã UTF-8 chuỗi ISO-8859-1	3
utf8_encode()	Mã hóa một chuỗi ISO-8859-1 sang UTF-8	3
xml_error_string()	Được một chuỗi lỗi từ các phân tích cú pháp XML	3
xml_get_current_byte_index()	Lấy các chỉ mục byte hiện tại từ các phân tích cú pháp XML	3
xml_get_current_column_number()	Được số cột hiện tại từ các phân tích cú pháp XML	3
xml_get_current_line_number()	Được số dòng hiện tại từ các phân tích cú pháp XML	3
xml_get_error_code()	Nhận được một mã lỗi từ các phân tích cú pháp XML	3
xml_parse()	Phân tích một tài liệu XML	3
xml_parse_into_struct()	Phân tích cú pháp XML dữ liệu vào một mảng	3
xml_parser_create_ns()	Tạo ra một phân tích cú pháp XML với sự hỗ trợ không gian tên	4
xml_parser_create()	Tạo một phân tích cú pháp XML	3
xml_parser_free()	Miễn phí một phân tích cú pháp XML	3
xml_parser_get_option()	Có được lựa chọn từ một phân tích cú pháp XML	3

<u>xml_parser_set_option ()</u>	Thiết lập các tùy chọn trong một phân tích cú pháp XML	3
<u>xml_set_character_data_handler ()</u>	Thiết lập chức năng xử lý dữ liệu nhân vật	3
<u>xml_set_default_handler ()</u>	Thiết lập mặc định chức năng xử lý	3
<u>xml_set_element_handler ()</u>	Đặt chức năng xử lý cho phần tử của các nguyên tố bắt đầu và kết thúc	3
<u>xml_set_end_namespace_decl_handler ()</u>	Thiết lập chức năng xử lý cho đến cuối tờ khai không gian tên	4
<u>xml_set_external_entity_ref_handler ()</u>	Đặt chức năng xử lý cho các thực thể bên ngoài	3
<u>xml_set_notation_decl_handler ()</u>	Thiết lập chức năng xử lý cho các tờ khai ký hiệu	3
<u>xml_set_object ()</u>	Sử dụng phân tích cú pháp XML trong một đối tượng	4
<u>xml_set_processing_instruction_handler ()</u>	Thiết lập chức năng xử lý để được hướng dẫn xử lý	3
<u>xml_set_start_namespace_decl_handler ()</u>	Thiết lập chức năng xử lý cho sự khởi đầu của việc kê khai không gian tên	4
<u>xml_set_unparsed_entity_decl_handler ()</u>	Thiết lập chức năng xử lý cho các tờ khai thực thể không được phân tách	3

Hằng số phân tích cú pháp XML PHP

Liên tục

XML_ERROR_NONE (số nguyên)

XML_ERROR_NO_MEMORY (số nguyên)

XML_ERROR_SYNTAX (số nguyên)

XML_ERROR_NO_ELEMENTS (số nguyên)

XML_ERROR_INVALID_TOKEN (số nguyên)

XML_ERROR_UNCLOSED_TOKEN (số nguyên)

XML_ERROR_PARTIAL_CHAR (số nguyên)

XML_ERROR_TAG_MISMATCH (số nguyên)

XML_ERROR_DUPLICATE_ATTRIBUTE (số nguyên)

XML_ERROR_JUNK_AFTER_DOC_ELEMENT (số nguyên)
XML_ERROR_PARAM_ENTITY_REF (số nguyên)
XML_ERROR_UNDEFINED_ENTITY (số nguyên)
XML_ERROR_RECURSIVE_ENTITY_REF (số nguyên)
XML_ERROR_ASYNC_ENTITY (số nguyên)
XML_ERROR_BAD_CHAR_REF (số nguyên)
XML_ERROR_BINARY_ENTITY_REF (số nguyên)
XML_ERROR_ATTRIBUTE_EXTERNAL_ENTITY_REF (số nguyên)
XML_ERROR_MISPLACED_XML_PI (số nguyên)
XML_ERROR_UNKNOWN_ENCODING (số nguyên)
XML_ERROR_INCORRECT_ENCODING (số nguyên)
XML_ERROR_UNCLOSED_CDATA_SECTION (số nguyên)
XML_ERROR_EXTERNAL_ENTITY_HANDLING (số nguyên)
XML_OPTION_CASE_FOLDING (số nguyên)
XML_OPTION_TARGET_ENCODING (số nguyên)
XML_OPTION_SKIP_TAGSTART (số nguyên)
XML_OPTION_SKIP_WHITE (số nguyên)

PHP Zip File Giới thiệu

Zip các tập tin chức năng cho phép bạn đọc các tập tin ZIP.

Cài đặt

Đối với các chức năng tập tin Zip để làm việc trên máy chủ của bạn phải được cài đặt, các thư viện:

- Thư viện ZLIB bởi Guido Draheim: [Tải về thư viện ZLIB](#)
- Zip PECL mở rộng: [Tải về Zip PECL mở rộng](#)

Cài đặt trên hệ thống Linux

PHP 5 +: chức năng Zip và thư viện Zip không được kích hoạt theo mặc định và phải được tải về từ các liên kết ở trên. Sử dụng - với-zip = DIR cấu hình tùy chọn để bao gồm hỗ trợ Zip.

Các hệ thống cài đặt trên

PHP 5 +: chức năng Zip không được kích hoạt theo mặc định, vì vậy php_zip.dll và thư viện ZLIBlib phải được tải về từ liên kết ở trên. php_zip.dll phải được kích hoạt bên trong php.ini.

Để kích hoạt bất kỳ phần mở rộng PHP, PHP extension_dir thiết lập (trong file php.ini) nên được đặt vào thư mục nơi mà các phần mở rộng PHP được đặt. Một giá trị extension_dir ví dụ là c: \ php \ ext.

Chức năng tập tin Zip PHP

PHP : cho biết phiên bản đầu tiên của PHP hỗ trợ chức năng.

Chức năng	Mô tả	P
<u>zip_close ()</u>	Đóng một file ZIP	4
<u>zip_entry_close ()</u>	Đóng một mục trong các file ZIP	4
<u>zip_entry_compressedsize ()</u>	Trả lại kích thước nén của một entry trong file ZIP	4
<u>zip_entry_compressionmethod ()</u>	Trả về phương pháp nén của một entry trong file ZIP	4
<u>zip_entry_filesize ()</u>	Trả về kích thước tập tin thực tế của một entry trong file ZIP	4
<u>zip_entry_name ()</u>	Trả về tên của một entry trong file ZIP	4
<u>zip_entry_open ()</u>	Mở một mục nhập trong tập tin ZIP để đọc	4
<u>zip_entry_read ()</u>	Đọc từ một mục mở trong tập tin ZIP	4
<u>zip_open ()</u>	Mở một tập tin ZIP	4
<u>zip_read ()</u>	Đọc vào tiếp theo trong một file ZIP	4

Chapter 31 : Linh Tinh Các Thứ Bên Ngoài Tổng Hợp Được

Code style theo chuẩn Zend

Tại sao chúng ta cần phải code theo chuẩn ?

Bởi vì chuẩn được đặt ra là để tối ưu hóa code của chúng ta, giúp code dễ đọc hơn, sau này dễ chỉnh sửa và khi làm việc nhóm thì điều này đặc biệt quan trọng !

PHP FILE FORMATTING

Tổng quát :

Trong những file chỉ chứa code PHP, tag đóng code `<?>` nên được bỏ qua . Bởi vì nó hok phải là 1 yêu cầu bắt buộc phải có của PHP và việc bỏ nó đi giúp hạn chế những khoảng trống có thể xuất ra trình duyệt .

Thực dòng :

Chúng ta cần thực dòng 4 khoảng trắng , hok được có tab

Số kí tự maximum của 1 dòng:

Số kí tự lớn nhất của 1 dòng trong PHP nên là 80 , mặc dù số kí tự lớn nhất thực sự có thể của 1 dòng PHP là 120

Kí tự xuống hàng

Mỗi dòng trong file phải được kết lại bằng kí tự `\n` theo kiểu Unix (tức là enter) .

Không được dùng carriage return (`\r`) <<<Mac dùng cái này

Không được dùng cả `\n` và `\r` (`\r\n`) <<< Window dùng cái này

1 dòng không nên có các khoảng trắng ở cuối dòng

QUY TẮC ĐẶT TÊN

CLASS: tên của class phải trở đến thư mục chứa nó

VD: trong thư mục tên Zend thì có class tên Zend , trong thư mục Zend/Db/Table.php có class tên Zend_Db_Table nằm trong file tên Table.php, chúng ta sẽ bàn về quy tắc đặt tên file sau.

Tên của class có thể chứa số nhưng không được khuyến khích . Và dấu gạch dưới (`_`) chỉ được dùng khi cần chia đường dẫn, như trường hợp Zend_Db_Table ở trên

Zend framework classes mà có tác giả là Zend hoặc các công ty hợp tác với Zend thì phải được đặt dưới thư mục Zend/ và tên class phải bắt đầu với Zend_

Lưu ý: nếu đoạn mã không phải 1 phần của framework, như là mã được viết bởi người dùng cuối, và hok phải Zend hay các công ty hợp tác với Zend thì không được phép đặt tên bắt đầu với : Zend_

Interfaces

Quy ước đặt tên Interface cũng giống với đặt tên class nhưng ở cuối tên phải thêm

_Interface

VD: Zend_Db_Table_Interface

Tên file :

Chỉ những kí tự, số, dấu gạch dưới (_) và dấu gạch ngang (-) được chấp nhận.

Các file chứa php code phải có extension là .php (trừ những file View được mặc định là .phtml)

Tên file phải tuân thủ việc trở đến các tên class như ở trên :

VD:

Zend/Db.php

Zend/Controller/Front.php

Hàm và phương pháp :

Tên hàm chỉ nên dùng kí tự, gạch dưới (_) không được chấp nhận . Trong tên hàm có thể có các con số nhưng không được khuyến khích

Tên hàm phải luôn luôn bắt đầu bởi các kí tự thường . Khi tên hàm gồm nhiều từ, kí tự đầu tiên của từ thứ 2 trở đi phải được viết hoa

Tên hàm có thể rườm rà, nhưng nó phải có nghĩa và dễ đoán được chức năng của nó

VD:

PHP Code:

```
filterInput()  
getElementById()  
widgetFactory()
```

Trong lập trình hướng đối tượng, các hàm truy cập vào đối tượng phải luôn được đặt tiền tố là “get” hay “set”

Mặc dù tên hàm hok thể dùng dấu gạch dưới, nhưng các method trong class được khai báo protected hay private, phải được bắt đầu với kí tự gạch dưới (_).

VD:

PHP Code:

```
class Zend_Foo  
{  
    protected function _fooBar()  
{  
    // Code ở đây  
}  
}
```

Hàm với tầm vực xuyên suốt script, có thể được sử dụng nhưng không khuyến khích . Các hàm này nên được đặt trong 1 class riêng biệt và khai báo static

Hàm hay biến với tầm vực static (trong 1 lớp) không nên khai báo private , thay vào đó là protected . Dùng final nếu hàm đó hok nên dc thừa kế .

Các tham số mặc định trong hàm

Sử dụng NULL thay cho FALSE nếu đặt mặc định

VD:

PHP Code:

```
Public function foo($required, $optional = NULL)
```

(chỉ khi tham số \$optional hok có giá trị mặc định ban đầu)

Tuy nhiên, nếu tham số mặc định có ý trả về true hay false, thì dùng false hay true lúc ấy mới được chấp nhận

Biến :

Tên biến có thể chứa các kí tự . Kí tự gạch dưới (_) hok được sử dụng, tên biến có thể sử dụng số nhưng hok được khuyến khích .

Với những biến nằm trong 1 class mà được khai báo private hay protected, kí tự đầu tiên của biến phải là 1 dấu gạch dưới (_) . Ngược lại, biến khai báo là public thì không được bắt đầu bằng 1 dấu gạch dưới (_) .

VD:

PHP Code:

```
class Zend_foo
{
    protected $_bar;
    public $bar;
}
```

Tương tự như tên hàm, tên biến phải luôn được bắt đầu bằng 1 kí tự thường .

VD:

PHP Code:

```
$nameOfCustomers
```

Tên của biến nên rườm rà, với điều kiện phải có nghĩa và đoán được chức năng của biến . Không nên đặt tên biến như \$i, \$n dùng để lặp các vòng lặp dù trong tình huống lặp ít nhất .

Với những vòng lặp trên 20 dòng code, tên biến dùng để đếm số lần lặp, phải có nghĩa .

Hằng :

Tên hằng có thể chứa kí tự và dấu gạch dưới và số .

Tên hằng phải được đặt theo quy luật : tất cả các kí tự đều viết hoa

Tên hằng phải dùng dấu gạch dưới (_) để cách các chữ có nghĩa

VD : 'VALUES_OF_PI' thay vì 'VALUESOFPI'

Giá trị Bool và NULL

Không giống các tài liệu PHP, Zend quy định kí tự thường cho cả 'true', 'false', và 'null'.

Coding Style

Tags :

PHP code phải được bao trong `<?php ?>` . Không được dùng dạng tắt : `<? ?>`

Chuỗi :

Chuỗi literal:

Khi chuỗi hok có biến ở trong (gọi là literal) , thì phải được khai báo với nháy đơn ' ' .

Chuỗi Literals có bao gồm nháy đơn bên trong :

Khi chuỗi có bao gồm nháy đơn ở trong, chúng ta phải khai báo nó bằng nháy đôi " " .

Nội suy biến :

2 dạng sau được chấp nhận trong việc nội suy biến :

PHP Code:

```
$greeting = "Hello $name, welcome back!";
```

```
$greeting = "Hello {$name}, welcome back!";
```

Nhưng dạng này thì không :

PHP Code:

```
$greeting = "Hello ${name}, welcome back!";
```

Lưu ý : Dấu { } sẽ hok được hiển thị ra browser

Nối chuỗi :

Khi nối chuỗi, 1 khoảng trắng phải được thêm vào trước và sau kí tự "." .

VD:

PHP Code:

```
$string = "$string1" . "$string2"
```

Hok hợp lệ:

PHP Code:

```
$string = "$string1"."$string2"
```

Khi thực hiện nối chuỗi, nên break lines

VD:

PHP Code:

```
$sql = "SELECT 'id', 'name' FROM 'people' "
      . "WHERE 'name' = 'Susan' "
      . "ORDER BY 'name' ASC ";
```

Mảng :

Mảng tuần tự :

Số âm không được chấp nhận trong việc đánh số thứ tự phần tử trong mảng .

Việc đánh số thứ tự của tất cả các mảng phải được bắt đầu bởi số 0 .

Khi khởi tạo mảng, 1 khoảng trắng cần phải được thêm vào sau mỗi dấu “,”.

VD:

PHP Code:

```
$sampleArray = array('first', 'second', 'third');
```

Hok hợp lệ:

PHP Code:

```
$sampleArray = array('first', 'second', 'third');
```

Mảng có thể được khởi tạo trên nhiều dòng . Và phải thêm khoảng trắng vào dòng thứ 2 trở đi để ngang hàng với dòng 1

VD:

PHP Code:

```
$sampleArray = array('first', 'second', 'third',  
                    'fourth', 'fifth', 'sixth',  
                    'seventh', 'eighth', 'ninth')
```

Mảng không tuần tự :

Khi khởi tạo mảng tuần tự, nên khai báo trên nhiều dòng , và mỗi dòng phải được thêm khoảng trắng từ dòng thứ 2 để ngang hàng với dòng thứ nhất

VD:

PHP Code:

```
$simpleArray = array('first' => 'Element1',  
                   'second' => 'Element2');
```

Lớp

Khai báo lớp

Kí tự mở class ({) phải được đặt ngay sau dòng khai báo tên class.

Mỗi dòng code bên trong class phải được thực dòng 4 khoảng trắng .

Chỉ 1 class trong 1 file PHP

Thêm code vào trong 1 file chứa class có thể được nhưng hok được khuyến khích

PHP Code:

```
/**  
 * class Docblock Here  
 */  
class Zend_Class  
{  
    // entire content of class  
    // must be indented four spaces  
}
```

Class member variables

Tên biến trong class ngoài việc được quy định theo cách đặt tên biến thông thường còn phải tuân theo các quy luật sau :

- Mỗi biến được khai báo trong class phải được đặt trên top
- Construct “var” không được dùng . Biến trong class phải được khai báo với các construct sau : private, protected, public . Truy cập trực tiếp vào biến trong class bằng cách set public không được khuyến khích, vì thế chúng ta cần có các hàm làm điều này và các hàm này phải có tiền tố là set hay get

Hàm và methods :

Các method phải được khai báo với các từ : private, protected, public. Không được bỏ trống

Static method phải được khai báo với theo độ ưu tiên của visibility trong script

VD:

PHP Code:

```
public    static foo() { ... }
private  static bar() { ... }
protected static goo() { ... }
```

Giống như class, kí tự mở hàm ({) phải được đặt ngay sau dòng khai báo tên hàm

VD:

PHP Code:

```
public function functionName()
{
    //code ở đây
}
```

Bên cạnh đó, không được phép có khoảng trắng giữa tên hàm và dấu (.

Hok hợp lệ:

PHP Code:

```
public function functionName ()
{
    //Code ở đây
}
```

Giá trị trả về trong hàm không được bao trong () .

VD:

PHP Code:

```
public function functionName()
{
    return $this->varName;
}
```

Hok hợp lệ:

PHP Code:

```
public function functionName()
{
    return ($this->varName);
}
```

Type hinting được khuyến khích sử dụng khi có thể

VD:

PHP Code:

```
class Zend_Component
{
    public function foo(SomeInterface $object)
    {
        .....
    }

    public function bar(array $options)
    {
        ....
    }
}
```

Hàm, method và sử dụng

Tham số của hàm phải được cách bởi 1 kí tự khoảng trắng . Ví dụ sau gọi hàm 3 tham số

VD:

PHP Code:

```
threeArguments(1, 2, 3);
```

Nếu muốn truyền tham trị, thì phải đặt & ở trước biến trong hàm khai báo , không được khi sử dụng hàm mới thêm dấu & .

Khi tham số của hàm có thể là mảng, khi gọi hàm có thể dùng array construct và có thể phân cắt mảng đó ra nhiều dòng

VD:

PHP Code:

```
threeArguments(array(1, 2, 3), 2, 3);
```

```
threeArguments(array(1, 2, 3, 'Zend', 'Studio',  
                    $a, $b, $c,  
                    56.44, $d, 500), 2, 3);
```

Câu lệnh điều khiển

If/ Else/ Elseif

Các câu điều kiện phải có 1 khoảng trắng ở giữa từ “if”, “elseif” và ngoặc mở “(“ . Thêm 1 khoảng trắng nữa giữa ngoặc đóng “)” và ngoặc cong “{“.

Ở trong ngoặc đơn “()”, giữa 2 bên của các toán tử so sánh phải có khoảng trắng để dễ đọc .

Ngoặc cong mở “{“ phải viết chung hàng với lệnh if . Ngoặc cong đóng phải ở hàng của riêng nó

VD:

PHP Code:

```
if ($a != 2) {  
    $a = 2;  
}
```

Hok hợp lệ:

PHP Code:

```
if($a!=2)  
{  
    $a=2;}
```

Với những lệnh if mà có thêm else, elseif thì phải tuân theo ví dụ sau :

PHP Code:

```
if ($a != 2) {  
    $a = 2;  
} else {  
    $a = 7;  
}
```

```
if ($a != 2) {  
    $a = 2;  
} else if ($a == 3) {  
    $a = 4;
```

```

} else {
    $a = 7;
}

```

Tất cả các lệnh if, elseif hay else phải có {} . Mặc dù trong vài trường hợp hok có chạy dc .

Phải dùng “elseif” , rất không khuyến khích dùng “else if”

Switch

Tương tự, giữa chữ “switch” và ngoặc mở “(“ phải có 1 khoảng trắng, và giữa ngoặc đóng “)” và “{“ phải có 1 khoảng trắng nữa .

Ngoặc cong mở “{“ phải nằm chung hàng với if , và ngoặc cong đóng “)” phải nằm ở hàng của riêng nó

Tất cả các nội dung bên trong switch phải được thực dòng 4 khoảng trắng . Và nội dung của mỗi “case” phải được thực dòng thêm 4 khoảng trắng nữa !

PHP Code:

```

switch ($numPeople) {
    case 1:
        break;

    case 2:
        break;

    default:
        break;
}

```

Và construct “default” hok được bỏ qua trong câu lệnh switch .

Lưu ý : Nếu trong 1 trường hợp nào đó, bạn cố tình bỏ break để code chạy 2 trường hợp trở lên, thì hãy comment tại case đó để báo là bỏ break có chủ đích chứ hok phải quên !

PHP Code:

```
// break intentionally omitted
```

Inline Documentation

Kí tự # hok nên dùng để comment trong code .

Files:

Trong mỗi file chứa PHP code, chúng ta phải có 1 block ở trên đầu file để miêu tả về file

.

Bạn có thể làm như sau :

PHP Code:

```

/**
 * Short description for file
 *
 * Long description for file (if any)...
 *
 * LICENSE: Some license information
 *
 * @copyright 2006 Zend Technologies
 * @license   [url]http://www.zend.com/license/3_0.txt[/url]   PHP License 3.0

```

```
* @version      $Id$
* @link         [url]http://dev.zend.com/package/PackageName[/url]
* @since        File available since Release 1.2.0
*/
```

Classes

Tương tự như vậy, các class cũng cần phải có

PHP Code:

```
/**
 * Short description for class
 *
 * Long description for class (if any)...
 *
 * @copyright     2006 Zend Technologies
 * @license       [url]http://www.zend.com/license/3_0.txt[/url]    PHP License 3.0
 * @version       Release: @package_version@
 * @link          [url]http://dev.zend.com/package/PackageName[/url]
 * @since         Class available since Release 1.2.0
 */
```

Functions

Với mỗi hàm, bao gồm cả các hàm dc định nghĩa trong class, cần có 1 block gồm:

- Miêu tả cơ bản về hàm
- Tất cả các biến của hàm
- Tất cả các giá trị trả về của hàm
- Nếu 1 hàm hay 1 method trong class có “throw exception” thì dùng @throw

VD:

PHP Code:

```
/**
 * Does something interesting
 *
 * @param Place $where Where something interesting takes place
 * @param integer $repeat How many times something interesting should happen
 * @throws Some_Exception_Class If something interesting cannot happen
 * @return Status
 */
public function doSomethingInteresting(Place $where, $repeat = 1)
{
    // implementation...
}
```

Require và include

Include, include_once, require, require_once hok nên dùng () .

VD :

PHP Code:

```
include 'header.php';
```

Hok hợp lệ:

PHP Code:

```
include ('header.php');
```

Những điều cần biết về PHP 5.4

PHP 5.4 là 1 phiên bản kế tiếp kể từ PHP 5.3 (hãy chuẩn bị cho sự ra đời PHP 6(hỗ trợ full Unicode) ngay từ bây giờ).

PHP 5.4 được tăng cường gỡ bỏ những hàm bị phản đối và tăng 20% tốc độ chạy, cùng với việc giảm bộ nhớ sử dụng !!!

Tìm hiểu về 1 số tính năng và cải tiến :

Một vài tính năng và cải tiến quan trọng nhất mà mình sẽ trình bày là : traits, cú pháp giản đơn về mảng, thiết lập webserver sử dụng với vài câu lệnh đơn giản, sử dụng \$this và truy cập các thành viên của class, kí hiệu <?= và 1 số tính năng khác sẽ xuất hiện trong PHP 5.4

PHP 5.4 cải thiện 1 cách đáng kể khả năng xử lý và sửa hơn 100 lỗi . Những tính năng đã được gỡ bỏ như : register_globals, magic_quotes (về thời gian) và safe_mode . Và 1 cải tiến giá trị khác là : khả năng hỗ trợ multibyte sẽ được bật mặc định và default_charset sẽ được thay đổi từ ISO-8859-1 to UTF-8

Content-Type: text/html; charset=utf-8 sẽ luôn được bao hàm . Vì thế không cần phải sử dụng nó cũng như thêm 1 số header để có được sự tương thích UTF-8.

Traits

Trait (horizontal reust/multible inheritance) là 1 bộ methods được cấu trúc như class (nhưng hok thể instantiate). Trait giúp các nhà phát triển sử dụng lại 1 bộ methods từ các class độc lập 1 cách tự do . Bởi vì PHP là 1 ngôn ngữ thừa kế đơn (single inheritance), 1 subclass có thể thừa kế từ chỉ 1 superclass, đó là nơi Trait phát huy tác dụng !

Ứng dụng lớn nhất của trait là khi nhiều class share cùng 1 hàm. Ví dụ, giả sử bạn đang build 1 website, và cần cả 2 Facebook và Twitter APIs . Chúng ta xây dựng 2 class có chung 1 function/method tên là cURL . Thay vì copy & paste đơn thuần method đó (để có thể sử dụng trong 2 class) , chúng ta sẽ sử dụng Trait (copy & paste, theo cách của compiler). Bằng cách này, chúng ta đã tạo ra 1 source code dễ sử dụng, và tuân thủ nguyên tắc DRY (Don't Repeat Yourself).

Ví dụ :

PHP Code:

```
/** cURL wrapper trait */
trait cURL
{
    public function curl($url)
    {
        $ch = curl_init();
```

```

        curl_setopt($ch, CURLOPT_URL, $url);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
        $output = curl_exec($ch);
        curl_close($ch);
        return $output;
    }
}

/** Twitter API Class */
class Twitter_API
{
    use cURL; // use trait here
    public function get($url)
    {
        return json_decode($this->curl('http://api.twitter.com/'. $url));
    }
}

/** Facebook API Class */
class Facebook_API
{
    use cURL; // and here
    public function get($url)
    {
        return json_decode($this->curl('http://graph.facebook.com/'. $url));
    }
}

$facebook = new Facebook_API();
echo $facebook->get('500058753')->name; // Xuất tên "Rasmus Lerdorf"

/** Bây giờ hãy tìm hiểu về sự tuyệt vời của cú pháp PHP 5.4 */
echo (new Facebook_API)->get('500058753')->name; // Xuất tên "Rasmus Lerdorf"
$foo = 'get';
echo (new Facebook_API)->$foo('500058753')->name; // và lại xuất tên "Rasmus Lerdorf"
echo (new Twitter_API)->get('1/users/show.json?screen_name=rasmus')->name; // và cũng là xuất Rasmus Lerdorf

```

Nếu cảm thấy khó hiểu, bạn có thể xem tiếp đoạn code sau :

PHP Code:

```

trait Net
{
    public function net()
    {
        return 'Net';
    }
}

trait Tuts
{
    public function tuts()
    {

```

```

        return 'Tuts';
    }
}

class NetTuts
{
    use Net, Tuts;
    public function plus()
    {
        return '+';
    }
}

$o = new NetTuts;
echo $o->net(), $o->tuts(), $o->plus();
echo (new NetTuts)->net(), (new NetTuts)->tuts(), (new NetTuts)->plus();

```

Lưu ý : magic constant trong trait là `__TRAIT__` .

Xây dựng CLI Webserver

Trong web development, người bạn tốt nhất của PHP là Apache HTTPD Server . Thỉnh thoảng, có thể là quá mức cần thiết khi cài đặt httpd.conf chỉ để sử dụng nó trong 1 môi trường phát triển web , khi bạn chỉ cần 1 web server nhỏ được bắt đầu với chỉ 1 dòng lệnh đơn giản ! Và vì thế PHP 5.4 đến với chức năng CLI Web server

Lưu ý : PHP Web server được xây dựng chỉ cho các nhà phát triển, hok được sử dụng trong mục đích thương mại !

Hướng dẫn ở phía dưới dành cho Window :

Bước 1 : tạo Document root directory, router file và index file .

Đến ổ hệ thống (ví dụ C:\). Tạo 1 folder tên public_html . Tạo 1 file trong folder này, đặt tên là router.php copy đoạn code dưới đây, và paste nó vào file vừa tạo :

PHP Code:

```

<?php
// router.php
if (preg_match('#\.php$#', $_SERVER['REQUEST_URI']))
{
    require basename($_SERVER['REQUEST_URI']); // serve php file
}
else if (strpos($_SERVER['REQUEST_URI'], '.') !== false)
{
    return false; // serve file as-is
}
?>

```

Bây giờ tạo 1 file khác tên index.php . Copy đoạn code sau và save vào file :

PHP Code:

```
<?php
// index.php
echo 'Hello Nettuts+ Readers!';
?>
```

Mở file php.ini (nằm ở trong đường dẫn PHP , ví dụ : C:\PHP)

Tìm include_path (nằm ở dòng thứ 708) . Thêm C:\public_html vào cuối chuỗi ở giữa nháy đôi, phân tách bởi dấu chấm phẩy (;) . Kết quả cuối cùng sẽ như thế này :

PHP Code:

```
include_path = ".;C:\php\PEAR;C:\public_html"
```

Save file lại và đến bước 2

Bước 2 : chạy web server .

Mở command prompt (Window + R, gõ cmd, nhấn enter) . Bạn sẽ thấy 1 cái gì đó đại loại như :

Mã:

```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
C:\Documents and Settings\nettuts>
```

Đổi đường dẫn hiện tại đến file cài đặt PHP bằng cú pháp sau :

Mã:

```
C:\Documents and Settings\nettuts>cd C:\php
C:\php>
```

Đến đoạn quan trọng nhất : chạy web server . Chép :

Mã:

```
php -S 0.0.0.0:8080 -t C:\public_html router.php
```

và paste vào command prompt (click chuột phải , chọn paste) . Nhấn enter . Nếu tất cả chạy đúng, bạn sẽ thấy 1 cái gì đó đại loại như bên dưới ^^ : (lưu ý hok được đóng command prompt, nếu đóng, web server sẽ bị tắt)

Mã:

```
C:\php>php -S 0.0.0.0:8080 -t C:\public_html router.php
PHP 5.4.0 Development Server started at Fri Mar 02 09:36:40 2012
Listening on 0.0.0.0:8080
Document root is C:\public_html
Press Ctrl-C to quit.
```

Mở browser lên và type vào :

Mã:

```
http://localhost:8080/index.php
```

bạn sẽ thấy :

Hello Nettuts+ Readers!

Lưu ý 1 : tạo 1 file php-server.bat với nội dung sau

Mã:

```
C:\php\php -S 0.0.0.0:8080 -t C:\public_html router.php.
```

Double click và chúng ta sẽ chạy được web server .

Lưu ý 2 : Sử dụng 0.0.0.0 thay vì localhost nếu bạn dự đoán server của bạn sẽ được truy

cập từ internet !

Cú pháp mảng ngắn gọn

PHP 5.4 cung cấp 1 cú pháp mảng ngắn gọn :

PHP Code:

```
$fruits = array('apples', 'oranges', 'bananas'); // "old" way

// Khai báo mảng như kiểu Javascript
$fruits = ['apples', 'oranges', 'bananas'];

// Mảng hok tuần tự
$array = [
    'foo' => 'bar',
    'bar' => 'foo'
];
```

Lưu ý : hãy nhớ rằng phương pháp cũ sẽ vẫn và mãi được sử dụng, cú pháp mảng ngắn gọn chỉ là 1 sự thay thế khi cần đơn giản !!!

Array dereference

Sẽ hok cần biến tạm khi xử lý với mảng .

Ví dụ chúng ta cần lấy tên ở giữa của chuỗi “Alan Mathison Turing “

PHP Code:

```
echo explode(' ', 'Alan Mathison Turing')[1]; // Xuất Mathison
```

Nhưng trước bản 5.4, chúng ta phải :

PHP Code:

```
$tmp = explode(' ', 'Alan Mathison Turing');
echo $tmp[1]; // Xuất Mathison
```

Bây giờ, nếu chúng ta muốn lấy từ Turing ?

PHP Code:

```
echo end(explode(' ', 'Alan Mathison Turing')); // Xuất Turing
```

Câu lệnh trên sẽ chạy tốt, nhưng sẽ hiện ra 1 lỗi E_STRICT (chuẩn Strict : chỉ biến mới được truyền tham trị) . Bởi vì nó trở thành 1 phần của E_ALL trong error-reporting.

Một ví dụ nâng cao :

PHP Code:

```
function foobar()
{
    return ['foo' => ['bar' => 'Hello']];
}
echo foobar()['foo']['bar']; // Hello
```

\$this trong các hàm vô danh

Bạn có thể trỏ đến các đối tượng từ các hàm vô danh (được biết với tên closures) bằng cách sử dụng \$this .

PHP Code:

```
class Foo
{
    function hello() {
        echo 'Hello Nettuts!';
    }

    function anonymous()
    {
        return function() {
            $this->
>hello(); // $this kiểu này sẽ hok chạy được trong các phiên bản trước
        };
    }
}
```

```
class Bar
{
    function __construct(Foo $o) // đối tượng của class Foo typehint
    {
        $x = $o->anonymous(); // lấy ra Foo::hello()
        $x(); // chạy Foo::hello()
    }
}
new Bar(new Foo); // Xuất Hello Nettuts!
```

Chúng ta vẫn có thể tiếp cận phương pháp này trong các phiên bản trước nhưng dùng câu lệnh quá mức cần thiết :

PHP Code:

```
function anonymous()
{
    $that = $this; // $that is now $this
    return function() use ($that) {
        $that->hello();
    };
}
```

Tag <?=>

Mặc kệ setting trong php.ini, short_open_tag , <?=> (open PHP tag và echo) sẽ luôn luôn available . Có nghĩa là bạn có thể an toàn dùng :

PHP Code:

```
<?=$title?>
```

Thay vì

PHP Code:

```
<?php echo $title ?>
```

Biểu diễn số nhị phân :

Kể từ bản này, số nguyên có thể là số thập phân (cơ số 10), số thập lục phân (cơ số 16), số bát phân (cơ số 8) hay nhị phân (cơ số 2) , được đặt các kí tự + - ở phía trước . Để ngụ ý số bát phân, đặt trước con số với 0 (zero). Để ngụ ý thập lục phân, đặt trước con số với 0x. Để ngụ ý nhị phân, đặt trước với 0b.

Ví dụ: để diễn tả số 31 chúng ta có

PHP Code:

```
echo 0b11111; // binary, introduced in PHP 5.4
echo 31; // duh
echo 0x1f; // hexadecimal
echo 037; // octal
```

Callable Typehint

Typehint được sử dụng để tăng cường sức mạnh của PHP . Typehint có thể là array, là object từ PHP 5.1 và là callable từ PHP 5.4 . Các kiểu typehint truyền thống : int hay string sẽ hok được hỗ trợ nữa !!

PHP Code:

```
function my_function(callable $x)
{
    return $x();
}

function my_callback_function(){return 'Hello Nettuts!';}

class Hello{static function hi(){return 'Hello Nettuts!';}}
class Hi{function hello(){return 'Hello Nettuts!';}}

echo my_function(function(){return 'Hello Nettuts!';}); // anonymous function
echo my_function('my_callback_function'); // callback function
echo my_function(['Hello', 'hi']); // class name, static method
echo my_function((new Hi), 'hello'); // class object, method name
```

Bộ đếm thời gian độ chính xác cao

`$_SERVER['REQUEST_TIME_FLOAT']` được thêm vào, với bộ đếm tính tới micro giây (theo kiểu float). Điều này rất quan trọng cho các bạn khi muốn tính tg chạy script .

PHP Code:

```
echo 'Executed in ', round(microtime(true) -
    $_SERVER['REQUEST_TIME_FLOAT'], 2), 's';
```

Lời kết

Về mặt tổng thể, PHP 5.4 cung cấp 1 số lượng lớn cải tiến .

Bạn nghĩ PHP 5.5 sẽ mang tới cho chúng ta cải tiến nào và bạn mong đợi điều gì ?

NGUỒN: Dịch từ <http://net.tutsplus.com/tutorials/php/you-must-know/> !

Các hàm rất hữu ích trong PHP bạn cần biết !

1/ Hàm có bao nhiêu tham số cũng được

Bạn có thể đã biết trong PHP chúng ta có thể tạo hàm với các tham số có giá trị mặc định ban đầu . Nhưng có 1 phương pháp cho phép 1 hàm chấp nhận mọi tham số !

Ví dụ đầu tiên với hàm có tham số có giá trị mặc định :

PHP Code:

```
// Hàm có 2 tham số mặc định
function foo($arg1 = '', $arg2 = '') {

    echo "arg1: $arg1\n";
    echo "arg2: $arg2\n";

}

foo('hello', 'world');
/* Xuất
arg1: hello
arg2: world
*/
```

```
foo();
/* Xuất
arg1:
arg2:
*/
```

Bây giờ, hãy cùng xây dựng 1 hàm có thể truyền vào mọi tham số . **func_get_args()** sẽ giúp chúng ta làm điều đó :

PHP Code:

```
// Tham số để rỗng nhé
function foo() {

    // returns an array of all passed arguments
    $args = func_get_args();

    foreach ($args as $k => $v) {
        echo "arg".($k+1).": $v\n";
    }

}

foo();
/* Không xuất gì cả */

foo('hello');
/* Xuất
arg1: hello
*/

foo('hello', 'world', 'again');
/* Xuất
arg1: hello
arg2: world
arg3: again
*/
```

2/ Sử dụng GLOB() để tìm files

Nhiều hàm trong PHP có tên dài và có tính mô tả chức năng của nó . Tuy nhiên, thật khó để nói hàm **glob()** có nhiệm vụ gì nếu chúng ta chưa từng sử dụng qua .

Thật ra, **glob()** có chức năng như hàm **scandir()** nhưng là 1 phiên bản cao cấp hơn ~

PHP Code:

```
// Lấy hết tất cả các file đuôi .php
$files = glob('*.php');
```

```
print_r($files);
/* Xuất
Array
(
    [0] => phptest.php
    [1] => pi.php
    [2] => post_output.php
    [3] => test.php
)
*/
```

Bên cạnh đó, chúng ta có thể lấy ra nhiều file :

PHP Code:

```
// Lấy tất cả file có đuôi .php và .txt
//GLOB_BRACE : ở trong {} sẽ dc hiểu là hoặc << hoặc php hoặc txt thì l
ấy ra
$files = glob('*. {php,txt}', GLOB_BRACE);
```

```
print_r($files);
/* Xuất
Array
(
    [0] => phptest.php
    [1] => pi.php
    [2] => post_output.php
    [3] => test.php
    [4] => log.txt
    [5] => test.txt
)
*/
```

File cũng có thể được lấy ra với đường dẫn :

PHP Code:

```
$files = glob('../images/a*.jpg');
```

```
print_r($files);
/* Xuất
Array
(
    [0] => ../images/apple.jpg
    [1] => ../images/art.jpg
)
*/
```

Còn nếu bạn muốn lấy đường dẫn đầy đủ, chỉ cần dùng **realpath()** :

PHP Code:

```
$files = glob('../images/a*.jpg');

// sử dụng hàm đến mỗi phần tử mảng
$files = array_map('realpath',$files);

print_r($files);
/* output looks like:
Array
(
    [0] => C:\wamp\www\images\apple.jpg
    [1] => C:\wamp\www\images\art.jpg
)
*/
```

Tất nhiên các ví dụ ở trên bạn test sẽ cho ra kết quả khác, vì folder trong máy của tác giả hok giống với folder của bạn . Tùy cơ ứng biến nhé ^^

3/ Thông tin bộ nhớ được sử dụng

Nếu biết được thời gian chạy script, bạn sẽ tối ưu code tốt hơn !

PHP có 1 bộ xử lý bộ nhớ rất phức tạp . Sự sử dụng bộ nhớ của đoạn script có thể tăng, giảm trong khi nó chạy . Để lấy bộ nhớ hiện tại đang được sử dụng, chúng ta dùng **memory_get_usage()** , để lấy bộ nhớ maximum được sử dụng trong cả quá trình, dùng **memory_get_peak_usage()** :

PHP Code:

```
echo "Ban đầu: ".memory_get_usage()." bytes \n";
/* prints
Ban đầu: 361400 bytes
*/

// Sử dụng bộ nhớ nào
for ($i = 0; $i < 100000; $i++) {
    $array []= md5($i);
}

// Loại bỏ mảng
for ($i = 0; $i < 100000; $i++) {
    unset($array[$i]);
}

echo "Cuối cùng : ".memory_get_usage()." bytes \n";
/* Xuất
Cuối cùng: 885912 bytes
*/

echo "Cao nhất: ".memory_get_peak_usage()." bytes \n";
/* Xuất
Cao nhất: 13687072 bytes
*/
```

4/Magic Constants

PHP cung cấp một số **magic_constants** hữu ích như :

lấy ra dòng hiện tại (**__FILE__**), lấy ra đường dẫn hiện tại (**__FILE__**), lấy ra thư mục (**__DIR__**), tên hàm (**__FUNCTION__**), tên lớp (**__CLASS__**), tên method (**__METHOD__**) và namespace (**__NAMESPACE__**)

Chúng ta sẽ hok đi từng constant, mà chỉ giới thiệu 1 vài ví dụ hữu ích :

PHP Code:

```
// đây là đường dẫn tương đối
// Có thể bạn sẽ gặp lỗi nếu chạy script từ thư mục khác đây nhé
require_once('config/database.php');
```

```
require_once(dirname(__FILE__) . '/config/database.php');
```

Bạn hãy chạy script trên, đừng tạo file database.php trong folder config nhé, để hiểu **__FILE__** có nhiệm vụ gì, bạn cứ việc copy->paste->chạy . Bạn sẽ thấy lỗi chưa ?
Nếu thấy rồi, bạn đảo thứ tự của 2 dòng **require_once()** trên, bạn chạy lại và sẽ hiểu ^^

Lưu ý : nên sử dụng **__FILE__** khi include script khác bạn nhé .

Sử dụng **__LINE__** sẽ giúp chúng ta debug dễ hơn . Bạn có thể tìm ra số thứ tự của dòng :

PHP Code:

```
// some code
// ...
my_debug("some debug message", __LINE__);
/* Xuất
Line 4: some debug message
*/

// some more code
// ...
my_debug("another debug message", __LINE__);
/* Xuất
Line 11: another debug message
*/

function my_debug($msg, $line) {
    echo "Line $line: $msg\n";
}
```

5/ Tạo ID đặc biệt :

Có 1 vài trường hợp chúng ta cần phải tạo ra những chuỗi đặc biệt (không trùng lặp), người ta thường dùng **md5()** để làm điều này, nhưng **md5()** không thực sự sinh ra để làm điều đó !

PHP Code:

```
// generate unique string
echo md5(time() . mt_rand(1,1000000));
```

Có 1 hàm chuyên dùng để xử lý điều đó trong PHP tên là : **uniqid()**

PHP Code:

```
// Tạo chuỗi đặc biệt
echo uniqid();
```

```

/* Xuất
4bd67c947233e
*/

// Tạo 1 chuỗi đặc biệt khác
echo uniqid();
/* Xuất
4bd67c9472340
*/

```

Có thể bạn đang thắc mắc rằng dù hàm tạo chuỗi đặc biệt nhưng có 1 số kí tự đầu tiên giống hệt nhau mỗi khi chạy ! Bởi vì việc tạo chuỗi này phụ thuộc vào thời gian server .

Để giảm cơ hội chúng có thể y chang nhau, chúng ta thêm prefix vào :

PHP Code:

```

// với prefix
echo uniqid('foo_');
/* Xuất
foo_4bd67d6cd8b8f
*/

// với nhiều kí tự hơn
echo uniqid(' ',true);
/* Xuất
4bd67d6cd8b926.12135106
*/

// prefix và nhiều kí tự hơn
echo uniqid('bar_',true);
/* Xuất
bar_4bd67da367b650.43684647
*/

```

Hàm này sẽ tạo chuỗi với số kí tự ít hơn **md5()**, sẽ giúp bạn tiết kiệm thời gian hơn .

6/ Serialization

Bạn đã bao giờ lưu 1 biến phức tạp vào database hay file ? Bạn không cần phải có những giải pháp của mình để chuyển đổi mảng về dạng chuỗi rồi nạp vào, bởi vì PHP đã có vào hàm dùng với mục đích này !

Chúng ta sẽ tìm hiểu 2 hàm : **serialize()** và **unserialize()**

PHP Code:

```

// 1 mảng phức tạp
$myvar = array(
    'hello',
    42,
    array(1, 'two'),
    'apple'
);

// convert to a string
$string = serialize($myvar);

```

```

echo $string;
/* Xuất
a:4:{i:0;s:5:"hello";i:1;i:42;i:2;a:2:{i:0;i:1;i:1;s:3:"two";}i:3;s:5:"
apple";}
*/

// Bạn có thể lấy lại mảng ban đầu
$newvar = unserialize($string);

print_r($newvar);
/* Xuất
Array
(
    [0] => hello
    [1] => 42
    [2] => Array
        (
            [0] => 1
            [1] => two
        )
    [3] => apple
)
*/

```

Tuy nhiên, từ khi JSON trở nên phổ biến, họ đã quyết định thêm sự hỗ trợ vào PHP5.2 .
 Bây giờ bạn có thể sử dụng **json_encode()** và **json_decode()** :

PHP Code:

```

// Mảng phức tạp
$myvar = array(
    'hello',
    42,
    array(1, 'two'),
    'apple'
);

// Chuyển đổi mảng thành string
$string = json_encode($myvar);

echo $string;
/* Xuất
["hello",42,[1,"two"],"apple"]
*/

// Lấy lại mảng ban đầu
$newvar = json_decode($string);

print_r($newvar);
/* Xuất
Array
(
    [0] => hello
    [1] => 42
    [2] => Array
        (

```

```

        [0] => 1
        [1] => two
    )

    [3] => apple
)
*/

```

Có vẻ 2 hàm này nhỏ gọn hơn nhĩ ^^ . Hơn thế nữa, chúng tương thích với javascript và 1 số ngôn ngữ khác . Tuy nhiên, cho những đối tượng phức tạp quá, có thể sẽ bị mất thông tin đấy !

7/ Nén chuỗi

Khi nói về sự nén, chúng ta cứ nghĩ là nén file zip !! Nhưng chúng ta hoàn toàn có thể nén 1 chuỗi phức tạp trong PHP .

Ở ví dụ sau đây, chúng ta sẽ dùng hàm **gzcompress()** và **gzuncompress()**

PHP Code:

```

$string =
"Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Nunc ut elit id mi ultricies
adipiscing. Nulla facilisi. Praesent pulvinar,
sapien vel feugiat vestibulum, nulla dui pretium orci,
non ultricies elit lacus quis ante. Lorem ipsum dolor
sit amet, consectetur adipiscing elit. Aliquam
pretium ullamcorper urna quis iaculis. Etiam ac massa
sed turpis tempor luctus. Curabitur sed nibh eu elit
mollis congue. Praesent ipsum diam, consectetur vitae
ornare a, aliquam a nunc. In id magna pellentesque
tellus posuere adipiscing. Sed non mi metus, at lacinia
augue. Sed magna nisi, ornare in mollis in, mollis
sed nunc. Etiam at justo in leo congue mollis.
Nullam in neque eget metus hendrerit scelerisque
eu non enim. Ut malesuada lacus eu nulla bibendum
id euismod urna sodales. ";

$compressed = gzcompress($string);

echo "Size ban đầu: ". strlen($string)."\n";
/* Xuất
Size ban đầu: 800
*/

echo "Size sau khi nén: ". strlen($compressed)."\n";
/* Xuất
Size sau khi nén: 418
*/

// Lấy lại chuỗi ban đầu
$original = gzuncompress($compressed);

```

Chúng ta có thể tiết kiệm tới 50% tổng kích thước đấy ! Bên cạnh đó, 2 hàm **gzencode()** và **gzdecode()** cũng làm điều tương tự nhưng dùng thuật toán khác thôi .

8/ Đăng kí 1 hàm shutdown

Với hàm **register_shutdown_function()**, chúng ta có thể làm 1 vài chuyện trước khi script ngừng chạy .

Ví dụ đếm thời gian chạy script nè :

PHP Code:

```
// Lấy thời gian bắt đầu chạy
$start_time = microtime(true);

// Code ở đây
// ...

// Xuất code đã chạy được bao lâu rồi
echo "Thời gian chạy script : ".
    (microtime(true) - $start_time).
    " seconds.";
```

Với cái nhìn đầu tiên, có thể bạn thấy thật tầm thường !! Đoạn code trên chỉ là thêm đoạn code tính thời gian ở cuối script . Tuy nhiên, nếu bạn gọi hàm **exit()** , đoạn code trên sẽ hok bao giờ chạy được . Thay vào đó, nó sẽ thông báo lỗi , hoặc là khi script bạn bị kết thúc bởi người dùng, bằng cách nhấn nút stop trên trình duyệt chẳng hạn !!!

Khi bạn sử dụng **register_shutdown_function()**, đoạn code của bạn sẽ chạy bất kể nó bị ngừng đột xuất vì lí do gì !

PHP Code:

```
$start_time = microtime(true);

register_shutdown_function('my_shutdown');

// Code ở đây
// ...

function my_shutdown() {
    global $start_time;

    echo "Thời gian chạy script : ".
        (microtime(true) - $start_time).
        " seconds.";
}
```

Nguồn : dịch từ <http://net.tutsplus.com/tutorials/php/need-to-know/> (đã test hết) + thêm vào 1 vài ý kiến riêng

P/s: có bỏ qua số 5/ vì nó hok sử dụng cho hệ điều hành WINDOW

Điều gì khiến bạn là 1 coder tồi ?

Tất cả chúng ta đều có những thói quen xấu, chúng ta sẽ tìm hiểu , cùng nhau kiểm tra, đánh giá lại và sửa ngay lập tức !

Bạn hok có kế hoạch trước khi bắt đầu code

Trước khi bạn bắt đầu code, bạn cần phải có 1 chiến lược . Điều này sẽ giúp bạn theo dõi được code, và tránh việc không hiểu code sẽ khiến bạn nhức đầu, ấy là chưa màn đến những linh hồn tội nghiệp khác đọc code của bạn =]]]

Một cách để khắc phục điều này là viết một đoạn phác thảo các chức năng của đoạn script bằng cách comment ở đầu file

Ví dụ :

PHP Code:

```
<?php
```

```
// Include necessary data

// Initialize the database connection

// Include the common header markup

// Determine the page variables from the POST data

// Load the proper database info using the page vairiables

// Loop through the loaded rows

    // Format the images for display

    // Create a permalink

    // Format the entry for display

    // Add the formatted entry to the entry array

// Collapse the entry array into page-ready markup

// Output the entries

// Include the common footer markup
```

Như bạn thấy, mặc dù chưa viết dòng code nào, chúng ta đã biết file này sẽ làm gì và nó sẽ như thế nào !! Và khi 1 chức năng nữa được thêm vào, bạn chỉ cần sửa comment ở trên .

Chúng ta cần phải thay đổi cách tiếp cận, nó sẽ giúp kĩ năng tổ chức code tốt hơn

Lưu ý : Đoạn comments trên chỉ là ví dụ . Một vài comments trên có thể hok cần thiết, 1 vài sẽ bị thay đổi, và 1 vài comments có các chức năng khác có thể sẽ được thêm vào file nữa .

Bạn không comment gì cả

Vấn đề tồi tệ nhất là khi chúng ta code, chúng ta không comment đầy đủ, hoặc thậm chí không comment .

Khi chúng ta cần phải chỉnh sửa code, có thể sẽ gây hỏng cả ứng dụng .

Quá trình comment code sẽ diễn ra trong khoảng từ 10 phút tới 6 giờ !

Ví dụ :

PHP Code:

```
$pieces = explode('.', $image_name);  
$extension = array_pop($pieces);
```

Đoạn code trên để làm gì ? Có phải bạn đang suy nghĩ để hiểu có nhiệm vụ gì ? Bạn vẫn đang hiểu 1 cách hok chắc chắn \$extension sẽ chứa gì ?

Vậy chúng ta thử comment :

PHP Code:

```
// Lấy ra định dạng của hình ảnh  
$pieces = explode('.', $image_name);  
$extension = array_pop($pieces);
```

Bây giờ thì không cần phải suy nghĩ gì cả : bạn nhìn comments, bạn nhìn tiếp code và hiểu ngay đoạn code có ý nghĩa gì !

Comment như trên có thể chỉ giúp bạn tiết kiệm thời gian 5 giây nhưng nếu trong 1 ứng dụng lớn, đó sẽ là 1 vấn đề !

Vậy, bây giờ hãy tập viết comment trước khi code đi nhé !!!

Bạn hi sinh sự rõ ràng để đổi lấy sự ngắn gọn

Một vài ví dụ điển hình cho việc hi sinh này là khi bạn dùng các biến với tên hok rõ ràng (như là \$a --- vậy \$a chứa gì ?) hay bỏ qua ngoặc cong { } .

PHP Code:

```
<?php
```

```
$foo = 8;
```

```
if( $foo<10 )  
    if( $foo>5 )  
        echo "Lớn hơn 5!";  
    else  
        echo "Nhỏ hơn 5!";  
else  
    echo "Lớn hơn 10!";  
    echo "<br />Another note.";
```

Với cái nhìn đầu tiên, bạn sẽ nghĩ dòng cuối cùng sẽ xuất ra nếu \$foo lớn hơn 10 ! Nhưng bởi vì dòng else và if nếu hok có ngoặc cong thì chỉ ảnh hưởng tới dòng ngay sau nó, nên

“Another note” sẽ bị bỏ qua .

Bạn có thể nhìn ra điều đó chỉ trong vài giây ? Tất nhiên là được

Nhưng chúng ta có nên lãng phí năng lượng để làm điều đó ? Hoàn toàn không

Thêm ngoặc cong vào chỉ làm code bạn tăng 1 vài dòng, nhưng nó sẽ giúp code rõ ràng hơn nhiều

PHP Code:

```
<?php

$foo = 8;

if( $foo<10 )
{
    if( $foo>5 )
    {
        echo "Lớn hơn 5 5!";
    }
    else
    {
        echo "Nhỏ hơn 5!";
    }
}
else
{
    echo "Lớn hơn 10!";
}
echo "<br />Another note.";
```

Vậy hãy cố gắng code thật rõ ràng, chỉ với vài dòng code bạn sẽ giúp ng khác không phải căng não ra để hiểu bạn đang code gì .

Bạn hok theo 1 chuẩn code nào cả

Viết code theo cách của bạn sẽ giúp bạn thỏa mãn, thấy dễ chịu nhưng đối với người khác thì hoàn toàn không .

Vì vậy, hãy viết code theo 1 chuẩn và duy trì điều đó !! Mọi ng sẽ cảm ơn bạn ! (Có thể bao gồm cả bạn đấy)

Bạn lặp lại code

Khi bạn quan sát code của bạn, nếu bạn gặp lỗi , hãy tự hỏi , mình có cần phải sửa nhiều hơn 1 file ? Nếu bạn trả lời có, đã đến lúc nhìn lại cách bạn code !!!

Nếu bạn có nhiều đoạn code cùng chức năng , bạn đang làm sai đấy !

Bạn không theo 1 mô hình nào cả

Code của bạn cần phải có 1 mô hình .Tôi không ngụ ý bạn nên dùng mô hình MVC hay

đại loại vậy. Ý tôi là bạn cần phải biết cách phân loại các thành phần, nội dung , v.v và những thành phần, nội dung ấy sẽ làm gì

Khi bạn code theo 1 mô hình, nhiều quyết định sẽ trở nên tự động hóa, và khi 1 người nào đó xem code bạn, họ sẽ hok phải đoán nhiều khi tìm kiếm 1 chức năng nào đó Làm điều này sẽ hok mất nhiều thời gian, nhưng sẽ giúp bạn rất nhiều đấy !

Vậy bạn hãy cố gắng code theo 1 mô hình nhé !!!!

Bạn quá thông minh cho các sản phẩm của bạn

Có 1 tí vấn đề về giải pháp đơn giản và giải pháp phức tạp

Chúng ta không cần 1 cách tiếp cận phức tạp với những vấn đề đơn giản !!!

Với 1 level cơ bản, giải pháp đơn giản luôn được mong đợi . Bạn đang đi từ điểm A đến điểm B, bạn đi đường vòng qua điểm C sẽ thú vị hơn, nhưng chẳng giúp ích được gì cả @@

Với 1 câu trả lời phức tạp, bạn sẽ khiến người khác không hiểu, không phải là vì ng khác hok thông minh như bạn, bởi có thể bạn học được từ 1 bài báo mà họ chưa đọc qua, và vì thế sẽ dẫn đến hiểu sai hoặc hok hiểu .

Đừng làm giảm giá trị của mình, nhưng cũng nhớ tránh những cách tiếp cận cực kì phức tạp với những vấn đề đơn giản nhé !

Bên cạnh đó, bạn hok nên code theo kiểu tự cho mình là 1 chuyên gia !! Với 1 triết lí đại loại là : nếu bạn hok hiểu code mình, tốt nhất đừng nên đọc nó .

Đây là 1 cách tiếp cận anti cộng đồng, và hok có tính xây dựng, hướng dẫn . Và 1 lập trình viên như vậy sẽ hok có sự giúp đỡ của mọi ng khi anh ta gặp vấn đề với code .

Vì vậy, tránh việc tạo code khó hiểu dưới mọi hình thức nhé . Điều đó sẽ hok khiến bạn thông minh hơn hay nhận được nhiều sự tôn trọng hơn đâu .

Không cập nhật kiến thức

Bên cạnh việc lười biếng comment, viết code khó hiểu, v.v ở trên thì việc hok cập nhật kiến thức cũng khiến bạn trở thành 1 coder tồi .

Công nghệ phát triển liên tục, công nghệ mới ra đời để giải quyết những vấn đề đang tồn đọng, nếu bạn chọn cách đứng yên không chịu tiến bộ thì thật là sai lầm trầm trọng .

Dưới đây là những vấn đề khác bạn nên tham khảo

Bạn cố gắng làm tất cả

Bạn không thể theo kịp cả cộng đồng . Nhưng thật may mắn, có 1 số người dành thời gian của họ để cập nhật công nghệ mới, và viết tutorial về những gì họ học được . Nếu

bạn tìm được trong những người này 1 người có cách tiếp cận và phong cách viết code giống bạn , bạn có thể học hỏi họ .

Việc theo dõi bài viết của vài người hợp với bạn sẽ tốt hơn nhiều so với theo dõi mỗi blog công nghệ bạn ghé qua với 1 vài lí do sau :

- Nếu bạn tin tưởng ý kiến của họ, họ sẽ giúp bạn học hỏi công nghệ
- Nếu 1 công nghệ xuất hiện nhiều lần trên các blog này, bạn hãy dành thời gian nắm bắt nó, vì nó sẽ là công nghệ phổ biến sau này
- Những tutorial của blog sẽ giúp bạn tiết kiệm rất nhiều thời gian khi phải bắt đầu từ con số 0

Bạn không dám đối mặt với thử thách

Nếu bạn không dám đối mặt với thử thách, thì bạn có vấn đề rồi đấy .

Hãy tìm 1 vài thử thách trong dự án bạn đang làm !

Tự hỏi bạn các câu sau khi đang xem dự án :

- Có công nghệ mới nào gây thích thú với mình được ứng dụng ở đây ?
- Mình đã học thêm được 1 công nghệ mới nào kể từ lần cuối cùng mình xem 1 dự án như thế này ?
- Đây là cách thực hiện tốt nhất để tôi có thể đảm bảo rằng tôi sẽ theo đuổi tới cùng project này ?

Lưu ý: tôi không nói về việc làm 1 cái gì đó rất phức tạp ở đây, có thể đơn giản chỉ là thêm Docblocks vào objects

Hãy cố không tự thuyết phục bản thân rằng mình đã học những gì cần học nhé !

Bạn hok chia sẻ

Hãy cải thiện bằng cách trao đổi, thảo luận với các đồng nghiệp coders nhé . Để làm được điều này có nhiều cách : Nếu bạn hiểu rõ 1 vấn đề nào đó, hãy viết tutorial về nó và cung cấp mã nguồn . Nếu bạn hok thích như thế, bạn có thể lướt qua vài diễn đàn và giúp đỡ những người mới .

Bạn sẽ hỏi : “Tại sao giúp đỡ những người mới sẽ giúp tôi tiến bộ ? “

Thông thường, nếu bạn post 1 giải pháp cần được tối ưu , sẽ có những lập trình viên chuyên nghiệp giúp đỡ bạn . Vậy chia sẻ là 1 mũi tên trúng 2 con nhạn : bạn không chỉ giúp đỡ người khác tiến bộ, mà các kĩ năng của bạn sẽ sắc bén hơn với sự trợ giúp của những cao thủ !

Bạn hok có những dự án thử nghiệm

Nếu chúng ta muốn tham gia vào 1 dự án lớn và hoành tráng, cách tốt nhất là chúng ta

hãy thử sức với những dự án thử nghiệm trước với các công nghệ đã biết .
Với phương pháp đó, chúng ta sẽ tiến bộ hơn, và không bao giờ trễ deadline hay làm sai cả .

Chúng ta đều có tội

Theo logic, nếu chúng ta tiến bộ vào thời điểm bây giờ, tức là trong quá khứ chúng ta đã rất tồi .

Tôi biết là khi tôi nhìn lại những đoạn code tôi viết trong quá khứ, tôi sẽ cảm thấy kinh hoàng !

Vì vậy....hãy dừng lại

Chúng ta đều không hoàn hảo . Nhưng chúng ta có thể làm mọi thứ với năng lực của mình để càng gần đích chúng ta muốn càng tốt

Các lỗi cơ bản trong lập trình PHP

Trong lập trình có 3 loại lỗi xếp theo thứ tự tăng dần độ khó của việc sửa lỗi là :

Lỗi cú pháp:

lỗi này dễ mắc phải nhưng cũng rất dễ sửa, vì khi PHP parse, lỗi sẽ được chỉ ra dòng mấy , lỗi gì . Chỉ cần chú ý kĩ lưỡng hơn thì sẽ tránh được lỗi này

Lỗi runtime

Lỗi này là lỗi về thời gian chạy script, ví dụ bạn gọi file : require('abc.php'). Nhưng file abc.php lại không tồn tại .

Lỗi logic

Đây là loại lỗi khó nuốt nhất và đau đầu nhất, vì khi chạy không hề có 1 lỗi nào cả, nhưng nó sẽ hok chạy như cách mà chúng ta muốn nó chạy @@ . Debug lỗi này là 1 quá trìnhchảy nước mắt .

Ví dụ đơn giản như bạn làm 1 phép toán cộng nhưng thay vì + bạn lại - => ra kết quả không mong muốn. Khi thuật toán càng phức tạp mà bị 1 lỗi logic thì có khi bỏ cả ngày để tìm xem nó sai ở đâu .

Dưới đây là 1 số lỗi hay thói quen hok tốt hay gặp :

1/ Nháy đơn, nháy đôi

Chúng ta thường hay sử dụng “ “ cho mọi chuỗi, dù là chuỗi đó có nội suy hay không. Điều này sẽ gây chậm cho code vì dấu nháy “ “ có nhiệm vụ truy xét xem có biến nào cần được nội suy hay không .

Lưu ý 1 : nháy đôi có thể chứa biến và nháy đơn, nháy đơn thì có thể chứa nháy đôi nhưng không thể xuất giá trị của biến .

Lưu ý 2 : biến môi trường không thể nội suy được như \$_POST, \$_GET, \$FILES, v.v

2/ Dấu chấm phẩy sau vòng lặp

Dấu chấm phẩy là 1 tác nhân cực kì quan trọng vì thế nên nó cũng có khả năng gây ra

những lỗi logic rất rất khó phát hiện .

Vd :

PHP Code:

```
for($i =0; $i<20; $i++) ;{
    echo $i;
}
//Chỉ xuất số 20 thay vì xuất từ 0->19 như bạn mong muốn .
// Vì for($i=0; $i<20; $i++); <<<<Lặp bản thân nó tăng $i cho đến khi $i=19 .
// Sau đó $i sẽ được tăng lên 1 là 20 rồi mới chạy xuống dòng echo $i
```

Bản chất của dòng FOR:

```
for($i = value1(1); $i < value2(2); $i++(3)){
code ở đây(4)
}
```

Dòng for trên sẽ làm công việc như sau : bắt đầu từ \$i = value 1, sau đó kiểm tra xem có value1 có bé hơn value2 hay không, nếu có, chạy code --->tăng \$i lên 1. Nếu không --->dừng lại .

Sơ đồ : (1) => (2) =>(4)=>(3)=>(1)=>....cho đến khi bước (1)=>(2) không thỏa nữa

3/ Thiếu break sau switch

Mọi người dùng switch thường rất hay quên break; sau mỗi trường hợp . Điều này gây ra 1 lỗi logic là code sẽ chạy nhiều hơn 1 trường hợp mà mình mong muốn.

4/ Không giới hạn thời gian chạy của script

Khi run 1 đoạn script rất hay xảy ra lỗi runtime, tức là script chạy gặp lỗi chạy hoài chạy hoài, hay cũng có thể là 1 vòng lặp đến vô cực

VD:

PHP Code:

```
$i =0;
while($i <3){
    echo "$i";
}
// Vòng lặp chạy mãi vì không tăng giá trị $i lên, $i mãi bằng 0 và nó mãi <3 ...
}
```

Như vậy sẽ rất tốn tài nguyên và thời gian, vì thế giới hạn 1 khoảng thời gian chạy script là 1 điều cần thiết .

Ta dùng hàm : set_time_limit(\$seconds) <<mặc định là 30s

Hàm set_time_limit khi được gọi nó sẽ reset lại bộ đếm, tức là nếu script tồn 2s để chạy đến dòng set_time_limit(1) thì tổng cộng cần 3 giây để script dừng lại .

Bạn có thể chỉnh giá trị mặc định default 'max_execution_time' trong file php.ini . Tuy nhiên 1 số host hok cho phép làm điều này . . .

ví dụ :

PHP Code:

```
set_time_limit(1);
$i =0;
while($i<3){
    echo 'a';
}
```

5/ Tin tưởng vào người dùng

Chúng ta thường dễ ý đến những trường hợp lớn mà bỏ qua các mắt xích nhỏ . Chúng ta thường quên xét trường hợp gặp người dùng nguy hiểm. Ví dụ tại field name của form, người ta nhập 1 các kí tự như ' " \$... Nếu chúng ta thao tác với những giá trị đó thì code sẽ vẫn chạy tốt nhưng khi validate thì sẽ có cả trăm lỗi về HTML => ảnh hưởng nghiêm trọng tới SEO .

Vì thế chúng ta nên dùng htmlentities() hay htmlspecialchars() để escape characters .

Cách dùng : \$name = htmlentities(\$_POST['name']);

//Hàm htmlentities() có nhiệm vụ chuyển đổi khá nhiều kí tự mặc định thành mã :

http://www.w3schools.com/html/html_entities.asp

//Hàm htmlspecialchars() có nhiệm vụ chỉ chuyển đổi 5 kí tự : > < ' " &

Chỉ khi click phải chọn view source thì mới thấy sự thay đổi của nó

Ví dụ: người dùng nhập vào field name : ABC ' s books & CD .

Sau đó : \$name = htmlentities(\$_POST['name'])

echo "\$name"; <<< Web vẫn hiển thị ABC ' s books nhưng khi view source thì là ABC ' s books & amp; CD

6/Sử dụng sai các toán tử

Hãy chắc chắn rằng bạn hiểu =, ==, === để sử dụng

= chỉ được dùng để gán biến và dùng trong query MySQL, == dùng trong if, v.v , còn === thì 2 giá trị phải bằng nhau và phải cùng kiểu dữ liệu .

Ví dụ 1 :

PHP Code:

```
$a = '5';  
if ($a == '6') {  
    code ở đây  
}
```

Ở trên ý đồ là nếu giá trị của \$a là 6 thì chạy code trong if, nhưng if(\$a == 'number') đúng với tất cả các giá trị number khác không

Ví dụ 2 : 5 == '5' là đúng nhưng 5 === '5' là sai

Trích dẫn: học hỏi từ 1 tài liệu tiếng anh, và viết lại theo phong cách của mình thêm vào 1 vài ví dụ

Code gửi mail trong PHP

Hàm mail cú pháp như sau : mail(to,subject,message,options)

Trong đó :

- + to: địa chỉ người nhận
- + subject :tiêu đề thư
- + message :nội dung thư
- + options:các tùy chọn bổ sung

Ví dụ :

File sendmail.php

PHP Code:

```
<?php
if (isset($_POST['sendmail']) && $_POST['sendmail']=="send")
{
    $from=$_POST['from'];
    $to=$_POST['to'];
    $subject=$_POST['subject'];
    $message=$_POST['message'];
    $options="Content-type:text/html;charset=utf-8\r\nFrom:$from\r\nReply-
to:$from";
    mail($to,$subject,$message,$options);
    echo"Đã gửi mail";
}
?>

<form method="post" action="sentmail.php">
Người gửi:<input type="text" name="from" size="70" /><br />
Gửi tới : <input type="text" name="to" size="70"/><br />
Tiêu đề :<input type="text" name="subject" size="70"/><br />
Nội dung :<br />
<textarea name="message" rows="10" cols="70">
</textarea><br />
<input type="submit" value="send" name="sendmail"/>
</form>
```

Xong !

Chúc thành công

Tự làm menu 2 cấp đơn giản với php và javascript

Bây giờ bắt đầu hén.

database đơn giản cho menu. khá đơn giản nên cũng không cần giải thích gì nhiều hén.
parent=0 là cấp cha. parent khác 0 chính là cấp cha của nó.

Kết nối database

PHP Code:

```
<?php
//=====Kết nối csdl =====
=====
    $host="localhost";
    $u="root";
    $p="root";
    $db="demo_db";
    $result=NULL;
    $conn=NULL;
    $conn=@mysql_connect($host,$u,$p) or die ("Ket noi server that bai"
);
    if($conn)
    {
        @mysql_select_db($db,$conn) or die("Ket noi database that bai")
;
    }
    mysql_query('SET CHARACTER SET utf8');
//=====
=====
?>
```

Sau đó đổ menu từ database ra ngoài (cả cha lẫn con)

PHP Code:

```
<?php
$sql="select * from cate_news where parent='0' order by date ASC";
$result=mysql_query($sql);
while($rows=mysql_fetch_array($result))
{
    ?>
    <!-- Main menu -->
    <div class="main_menu">
        <a style="width:100%; height:100%;" href="index.php?cate=<?php echo
        $rows['id']; ?>"><div style="line-height:30px; padding-
        left:10px; background:#006699; width:200px; border-
        bottom:1px solid #FFFFFF;"><?php echo $rows['name']; ?></div></a>
    </div>
    <!-- Hết main menu -->
    <!-- Sub menu -->
    <div style="margin-
    left:30px;" class="sub_menu" id="sub_<?php echo $rows['id']; ?>">
        <?php
            $sql2="select * from cate_news where parent='".$rows['id'].
            "' order by date ASC";
            $result2=mysql_query($sql2);
```

```

        while($rows2=mysql_fetch_array($result2))
        {
            ?>
            <div><a href="index.php?cate=<?php echo $rows2['id']; ?>">
- <?php echo $rows2['name']; ?></a></div>
            <?php
            }
        ?>
    </div>
    <!-- Hết Sub menu -->
    <?php
    }
?>

```

Tiếp theo là ẩn hết menu con đi

HTML Code:

```

.sub_menu
{
    margin-top:5px;
    margin-bottom:5px;
    display:none;
}

```

Cuối cùng là cho nó bung sub menu ra theo điều kiện. dùng javascript

HTML Code:

```

<!-- Xỏ sub menu -->
<script>
    var id='<?php echo $_GET['cate']; ?>';
    var sub_menu='sub_'+id;
    if(document.getElementById(sub_menu))
        document.getElementById(sub_menu).style.display='block';
    else
    {
        <?php
        $cate=$_GET['cate'];
        $sql="select * from cate_news where id='".$cate."'";
        $result=mysql_query($sql);
        if(mysql_num_rows($result)>0)
        {
            $rows=mysql_fetch_array($result);
            ?>
            var id='<?php echo $rows['parent']; ?>';
            var sub_menu='sub_'+id;
            if(document.getElementById(sub_menu))

            document.getElementById(sub_menu).style.display='block';
            <?php

```

```

    }
    ?>

}
</script>

```

Tự làm điểm sáng với javascript

cái này trên mạng có code sẵn và viết bằng jquery, nhưng mà mình tự viết lại vì mình viết mình sẽ hiểu rõ và chỉnh sửa dễ dàng hơn rất nhiều, và hiểu rõ được một số cái mới nữa.

Thật sự nó chỉ là cái div nổi lên trên mà thôi

có các đặc điểm sau:

- Khi vào trang index hiện ra 1 pop-up quảng cáo nhỏ, nằm giữa trang web.
- Tự tắt sau 5s.
- Có nút Close để tắt.
- Làm mờ background trang web.

DEMO

Đầu tiên là định dạng

HTML Code:

```

<style>
#popup
{
    position:fixed;
    z-index:99999;
    background:#FFFFFF;
    border:5px solid #003300;
    padding:5px 0px 5px 5px;
    display:none;
}
#bg_trans
{
    position:fixed;
    z-index:99998;
    display:none;
    background-color:#000000;
    opacity:0.4;filter: alpha(opacity=40);
    -ms-
filter:"progid:DXImageTransform.Microsoft.Alpha(Opacity=40)";
}
</style>

```

Đặt thuộc tính onload hoặc là gọi làm đặt ở cuối trang web cũng không sao

HTML Code:

```

<body topmargin="0" leftmargin="0" onload="show_popup();" >

```

- cái popup và nền

HTML Code:

```

<div id="popup" style="width:200px;">

```

```

--> <!-- //////////////////////////////////////
--> <!-- Nút close -->
--> <!-- //////////////////////////////////////
--> <div style="width:27px; right:0px; top:0px; height:18px;
position:absolute; z-index:999999; background:url(x.jpg);
cursor:pointer;" onclick="hide_popup();"></div>
--> <!-- hết nút close //////////////////////////////////
--> <!-- Nội dung ở đây -->
--> <!-- //////////////////////////////////
nội dung popup nội dung pop-up <br />
nội dung popup nội dung pop-up<br />
nội dung popup nội dung pop-up<br />
nội dung popup nội dung pop-up<br />
nội dung popup nội dung pop-up<br />
nội dung popup nội dung pop-up<br />
nội dung popup nội dung pop-up<br />
nội dung popup nội dung pop-up<br />
nội dung popup nội dung pop-up<br />
nội dung popup nội dung pop-up<br />
--> <!-- hết nội dung //////////////////////////////////
-->
</div>
<div id="bg_trans">
</div>

```

HTML Code:

```

<script>
    var time=5; // thời gian
    time=(time*10)-10;
    var count=0;
    var type;
    type=navigator.appName;
    function show_popup()
    {

        count++;
        if(type=="Microsoft Internet Explorer")
        {
            if (document.compatMode=="BackCompat")
            {
                var height=document.body.offsetHeight;
            }
            else
            {
                var
height=document.documentElement.offsetHeight;
            }
        }
        else
        {
            var height=window.innerHeight;
        }
        if (document.compatMode=="BackCompat")
            var width=document.body.offsetWidth;
    }

```

```

else
    var width=document.documentElement.offsetWidth;

document.getElementById('popup').style.display='block';

document.getElementById('bg_trans').style.display='block';

document.getElementById('bg_trans').style.width=width+'px';

document.getElementById('bg_trans').style.height=height+'px';
    var
t_width=document.getElementById('popup').offsetWidth;
    var
t_height=document.getElementById('popup').offsetHeight;

document.getElementById('bg_trans').style.top="0px";

document.getElementById('bg_trans').style.left="0px";
    var mar_top=(height/2)-(t_height/2);
    var mar_left=(width/2)-(t_width/2);

document.getElementById('popup').style.top=mar_top+"px";

document.getElementById('popup').style.left=mar_left+"px";
    if(count<=time)
    {
        setTimeout("show_popup()",100);
    }
    else
    {

document.getElementById('popup').style.display='none';

document.getElementById('bg_trans').style.display='none';
    }
}
function hide_popup()
{
    time=0;
    document.getElementById('popup').style.display='none';

    document.getElementById('bg_trans').style.display='none';
}
</script>

```

Mình đặt hàm setTimeout cho cả khối xử lý pop-up vì mình muốn nếu người dùng kéo cửa sổ thu nhỏ hay mở rộng, pop-up tự động chạy theo

Vì là javascript nên các bạn chỉ cần view->source hoặc save as trang web về là có hết code rồi.

Các bạn có thể ứng dụng để làm code quảng cáo 2 biên trang web hoặc lightbox, tất nhiên là không dùng jquery nên hiệu ứng nó không mượt như dùng jquery.

Giới thiệu - Cài đặt Smarty

Thông thường khi lập trình PHP bạn thường hay viết lẫn cả phần HTML với mã lệnh PHP trên một trang. Điều này sẽ dẫn đến rất rối nhất là khi bảo trì. Hoặc là 1 ngày đẹp trời khách hàng của bạn không thích giao diện của website hiện tại, họ đòi hỏi phải thay giao diện khác. Bạn là người viết ra và cũng là người bảo trì thì không sao nhưng nếu bạn viết ra sau đó người khác bảo trì, việc đọc lại source code là 1 điều hết sức khó khăn. Hơn nữa, việc chỉnh sửa template theo cách viết thông thường rất mất thời gian.

Vậy làm sao để giải quyết vấn đề trên: làm thế nào để độc lập mã PHP và mã HTML trên 1 website hay nói cách khác làm sao để tách biệt được việc thiết kế và lập trình website? Một website ra đời là sự kết hợp giữa thiết kế đồ họa và lập trình, làm sao để việc thiết kế không ảnh hưởng đến lập trình hay nói cách khác khi tạo ra 1 giao diện mới thì người lập trình không phải thay đổi phần chức năng bên lập trình PHP của mình

Vì lí nhiều do này nhiều template ra đời như Smarty, PHP Savant, PHPLib, Yats, FastTemplate,... trong đó nổi bật nhất là Smarty và PHP Savant.

Vậy các template engine hoạt động như thế nào, bạn hãy xem topic sau: (link)

Tại sao ta lại chọn template engine Smarty?

- Thứ nhất vì nó dễ sử dụng, dung lượng nhẹ
- Hỗ trợ cache giúp tải trang nhanh hơn
- Được hỗ trợ tài liệu dễ đọc
- Được xây dựng trên kiến trúc 3 tầng

Presentation Logic Layer : Nơi đặt bố cục Website , trình diễn dữ liệu được lấy ra từ Business Logic Layer:

+CSS

+Image

+JavaScript

+XML

+RSS

Business Logic Layer : Các Nghiệp Vụ trên hệ Thống

Data Access Layer : Các Class tương tác với Database

Presentation Logic Layer <=> Business Logic Layer <=> Data Access Layer

- Hỗ trợ các hàm cho hệ thống riêng trong template
- Cung cấp trình debug console
- Plugins: cung cấp cho việc sử dụng lại code
- Filters, Smarty Internals,...

Chúng ta bắt đầu làm cài đặt và làm 1 ví dụ đơn giản nhé!

Bước 1: bạn download library Smarty từ địa chỉ: <http://www.smarty.net/download>
Hiện tại version mới nhất là 3.1.5
Sau khi download bạn giải nén

Bước 2: Tạo các thư mục cho project

- Bạn tạo 4 thư mục: templates, templates_c, cache, includes trong www/myWebsite/
- Tạo thư mục template: www/myWebsite/templates/default/ chứa giao diện
- Bạn copy thư mục libs (Smarty vừa download) vào root/www/myWebsite/includes và rename thành "smarty" nhé!

- Ta tạo file cấu hình "config.php" trong thư mục /includes cho project, file này khởi tạo Smarty, cấu hình database,...
 - Tạo file index.php trong www/myWebsite, và template của nó là index.html trong myWebsite/templates/default/
- Như vậy cấu trúc các thư mục sau khi tạo như sau:

Bước 3: Cấu hình khởi tạo Smarty

- Bạn mở includes/config.php và paste code cấu hình sau:
- Thay đổi \$path_site nhé

PHP Code:

```
// Config pages
$path_site = "/myWebiste/"; // đường dẫn thư mục website, slash đầu và
                             cuối dòng
define('_BASE', $_SERVER['DOCUMENT_ROOT'].$path_site); // đường dẫn t
uyệt đối E:/Wamp/www/myWebsite/
// Smarty config
include _BASE.'includes/smarty/Smarty.class.php';

$smarty = new Smarty;

$smarty-
>template_dir = _BASE.'templates/default/'; // template folder
$smarty->compile_dir = _BASE.'templates_c/'; // template cache
$smarty->cache_dir = _BASE.'cache/'; // php cache
$smarty->config_dir = _BASE.'includes/smarty/config/'; // smarty cache
$smarty->compile = true; // end able compiler, debug
```

Bước 4: Xong cấu hình, ta thử tạo 1 template đơn giản nhé!

Template hiển thị: "Chào mừng bạn: Phú"; với giá trị "Phú" là 1 giá trị được tạo ra từ biến \$name trong index.php

- Mở myWebsite/index.php

PHP Code:

```
// include file config
require 'includes/config.php';
$name = "Phú"; // tạo biến
$smarty->assign("person", $name); // nạp $name vào template, tạo ra biến person cho template sử dụng
$smarty->display("index.html"); // nạp template index.html
?>
```

- Mở myWebsite/templates/default/index.html và tạo ra 1 template đơn giản

PHP Code:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Hello World</title>
</head>

<body>
    <h1>Chào mừng bạn: {$person} !</h1>
</body>
</html>
```

- Chạy <http://localhost/myWebsite/index.php> và xem kết quả

Smarty cơ bản [Phần 1]

BÀI 1: HIỂN THỊ DỮ LIỆU TỪ MYSQL BẰNG SMARTY

1.Hình ảnh bài tập

Firefox ▾

localhost / localhost / student / stud... ×

THÔNG TIN HỌC SINH × +

localhost/smarty/go/

Disable ▾

Cookies ▾

CSS ▾

Forms ▾

Images ▾

Information ▾

Miscellaneous ▾

Outline ▾

Tài khoản của TuanBk		
Tên:	Tuan	
Ngày sinh	26-11-1991	
Nhóm	B+	
Email	tuanbk.spam@gmail.com	

Tài khoản của Maria Bule		
Tên:	L ♦ Thu Trang	
Ngày sinh	10-04-1994	
Nhóm	B+	
Email	jungok@yahoo.com	

:2.Hướng dẫn cài đặt smarty:

3. Hướng dẫn giải quyết bài tập

- Công việc đầu tiên ta sẽ phải tạo 3 folder gồm:
 - + Cache: Dùng để lưu dữ liệu giúp website truy xuất nhanh hơn
 - + Tempalates: Dùng để lưu các file templates dạng .tpl
 - + Configs: Dùng để lưu file .conf
 - +Tempalates_c: Dùng để lưu trữ những phát sinh lỗi của hệ thống.
- Tạo cơ sở dữ liệu trong MySQL

+ Database: student

+Các trường: id_student: AI, Primary,INT.

nick: Varchar 100.

name: Varchar 100.

ngaysinh: Varchar 100.

nhom: Varchar 100.

email: Varchar 100.

- Viết code

Tạo file index.php

Với dạng code như sau

PHP Code:

```
<?php
require('../libs/Smarty.class.php'); // Đây là thư viện của hệ thống

$smarty= new Smarty(); // Khởi tạo Smarty
$con=mysql_connect("localhost","root","mysql") or die ("Don't cant data
base");
// Kết nối tới MySQL của bạn
mysql_select_db("student",$con) or die ("Don't cant database ");
// Kết nối tới Database
$result=mysql_query("SELECT * FROM student_database WHERE 1");
//Chọn tất cả các trường trong cột student_database.
$students=array();
//Khởi tạo biến $student dạng mảng để gán dữ liệu
while ($row=mysql_fetch_assoc($result)) // Lấy từng bản ghi
{
    $students[]=$row; // Gán biến mảng với từng dữ liệu
}
$smarty->
    assign("students",$students); // Gán dữ liệu cho biến mảng Smarty
$smarty->
    display("index.tpl"); // Hiển thị ra ngoài màn hình với file index.tpl
?>
```

Sau đó chúng ta sẽ tạo 1 file index.tpl trong thư mục templates với dạng code như sau:

PHP Code:

```
<html>
<title> THÔNG TIN HỌC SINH </title>
<head>
</head>
<body>
{section name=st loop=$students} // Khởi tạo Section lấy biến Smarty $s
tudents khi chúng ta gán $smarty->assign("students",$students);
<table width="80%" border="1" cellpadding="4">
```

```

<tr>
<td width="38%" bgcolor="#CCCCCC">
<strong>Tài khoản của &nbsp;   {$students[st].nick}</strong>
// Lấy dữ liệu của trường nick tương ứng trong MYSQL
</td>
<td width="62%">&nbsp;   </td>
</tr>
<tr>
<td><font><b>Tên: </b></font></td>
<td><font><b><bgcolor="#CCCCCC">{$students[st].name}</b></font></td>
</tr>
<tr>
<td>Ngày sinh</td>
<td>{$students[st].ngaysinh}</td>
</tr>
<tr>
<td>Nhóm</td>
<td>{$students[st].nhom}</td>
</tr>
<tr>
<td>Email</td>
<td>{$students[st].email}</td>
</tr>
</table>
<br />
</section>
</body>
</html>

```

Bước 4: Test khởi chạy

Chúng ta sẽ chạy thử xem có lỗi gì không:

<http://localhost/smarty/go>

Chú ý: Ở đây mình tạo một thư mục tên go nằm trong nó có tất cả folder và file trên.

Smarty cơ bản [Phần 2]

Xin chào các bạn ở phần I các bạn đã được nắm được cách cài đặt smarty và khởi chạy chương trình với bài tập show dữ liệu ra bên ngoài

Ở trong phần II bài tập sẽ khó hơn một chút, các bạn sẽ phải tạo liên kết giữa 3 bảng và show nó ra ngoài màn hình, ngoài ra ở phần này mình sẽ đá thêm tạo style cơ bản cho các bạn.

Bài tập:

area	agent_name	item_name	item_price	quantity	total
Ha Noi	Quan Thanh Xuan	Iphone 5	5	16000000	80000000
Ninh Binh	Thi Xa Tam Diep	Laptop Acer V3-471	10	11000000	110000000



1. Tạo liên kết 2 bảng xuất ra màn hình và sửa style cơ bản cho bảng.

2. Hướng dẫn cho bảng.

Như ở bài trước mình đã hướng dẫn các bạn cài đặt smarty, bạn nào chưa hiểu thì đọc lại bài trước nhé.

Ở bài này mình sẽ hướng dẫn các bạn nối 2 bảng với nhau với câu lệnh INNER JOIN tạo sự liên kết giữa các bạn

Và đá qua style cơ bản một tí cho các bạn nắm được thao tác cơ bản để style giao diện trên smarty

Ok, lets go

Bước 1:

Tạo cơ sở dữ liệu SQL

Bước này thì ai cũng đã làm đc rồi, các bạn tạo cơ sở dữ liệu như sau:

Database: daily_banhang

Bảng:

Khuvuc: ten_khuvuc: Varchar 50

Id_khuvuc: int 11

Ten_daily: varchar 50

Mathang: id_mathang: int 11

Ten_mathang: varchar 500

Soluong: int 100

Ketqua: id_kv: int 11

Id_mathang: int11

Dongia: int

Bước 2: Viết code nào

Các bạn tạo một file có tên daily_banhang.php

Đoạn mã như sau:

PHP Code:

```
<?php
include("../libs/smarty.class.php");
$smarty = new smarty();
mysql_connect("localhost","root","mysql") or die ("Lỗi không thể kết nối tới CSDL");//Dòng này là kết nối CSDL của các bạn
mysql_select_db("daily_banhang") or die ("Lỗi không thể kết nối tới Database");
// Dòng này là kết nối tới Database
$sql=mysql_query("SELECT      ten_khuvuc,
ten_daily,
ten_mathang,
dongia,
soluong,
(dongia*soluong) as tong_tien
FROM khuvuc
INNER JOIN ketqua on khuvuc.id_khuvuc=ketqua.id_khuvuc
```

```

INNER JOIN mathang on ketqua.id_mathang=mathang.id_mathang
ORDER BY ten_khuvuc ASC");
// Chọn CSDL kết nối 2 bảng bằng INNER JOIN
$khuvuc=array();
$daily=array();
$mathang=array();
$soluong=array();
$dongia=array();
$tong=array();
while ($row=mysql_fetch_assoc($sql))
{
$khuvuc[]=$row['ten_khuvuc'];//Gán trường tên khu vực vào biến khu vực
$daily[]=$row['ten_daily'];
$mathang[]=$row['ten_mathang'];
$soluong[]=$row['soluong'];
$dongia[]=$row['dongia'];
$tong[]=$row['tong_tien'];
}
$tieude=array("Khu vực","Tên đại lý","Tên mặt hàng","Số lượng","Đơn giá",
,"Tông tiền");// Tiêu đề của bảng
$smarty->assign("khuvuc",$khuvuc);
$smarty->assign("daily", $daily);
$smarty->assign("mathang", $mathang);
$smarty->assign("soluong", $soluong);
$smarty->assign("dongia", $dongia);
$smarty->assign("tong", $tong);
$smarty->assign("tieude", $tieude);
$smarty->display("daily_banhang.tpl");// In ra màn hình
mysql_free_result($sql);
?>

```

Tiếp theo ta tạo thêm một file tên daily_banhang.tpl ở trong thư mục templates

Code như sau

PHP Code:

```

<html>
<title> DAI LY BAN HANG </title>
<head>
</head>
<body>
<table cellpadding="4" border="1" cellspacing="0">
<tr>
{section name=id loop=$tieude }
<th id="cottieude">{$tieude[id]}</th>
{/section}
</tr>
{section name=data loop=$khuvuc}
<tr id="cotbang">
<td>{$khuvuc[data]}</td>
<td>{$daily[data]}</td>
<td>{$mathang[data]}</td>
<td>{$soluong[data]}</td>
<td>{$dongia[data]}</td>
<td>{$tong[data]}</td>
</tr>
{/section}
</table>
</body>

```

```
</html>
<style>
#cottieude{
text-transform: uppercase;
background-color:#25510D;
color: #EEEEEC;
}
#cotbang{
background-color:#ACABAB;
text-weight:bold;
text-align:center;
}</style>
```

Mình giải thích thêm đoạn style ở đây mình có thể tạo style ở bất cứ đâu bạn chỉ cần chèn vào

PHP Code:

```
<th id="cottieude"></th>
```

Nhìn thế này chắc các bạn cũng hiểu. Chúc các bạn thành công

Kỹ thuật sử dụng lazy loading trong hướng đối tượng

Ở các bài trước, chúng ta đã tìm hiểu về các [kiến thức cơ bản trong lập trình hướng đối tượng](#). Tiếp tục trong bài này, chúng ta sẽ cùng tìm hiểu về kỹ thuật sử dụng **lazy loading** trong ứng dụng.

Lazy loading là kỹ thuật rất quan trọng trong lập trình hướng đối tượng. Vì nhờ có nó ta có thể tiết kiệm được rất nhiều thao tác gọi nạp file khi sử dụng. Bản thân **lazy loading** sử dụng magic method (các phương thức đặc biệt trong PHP) để triệu gọi dữ liệu. Cụ thể là phương thức `__autoload()`.

Phương thức này sẽ được tự động gọi khi một đối tượng được khởi tạo. Cũng vì lý do đó mà chúng ta sử dụng nó để `require()` một file chứa lớp đó.

Ví dụ:

Tạo 1 file tên `test.php` với nội dung sau:

PHP Code:

```
<?php
function __autoload($url) {
    require("$url.php");
}
$abc=new ABC;
```

```
$abc->demo();  
?>
```

Tiếp tục ta tạo 1 file tên ABC.php với nội dung sau:

PHP Code:

```
<?php  
class ABC{  
    public function demo(){  
        echo "<h2>Welcome to QHOnline Tutorial</h2>";  
    }  
}
```

Khi chạy file test.php thì kết quả sẽ hiển thị trên màn hình là:

Welcome to QHOnline Tutorial

Rõ ràng trên file test.php của chúng ta ở trên không hề có class nào tên ABC và cũng không có phương thức nào tên là demo() cả. Vậy thì vì sao ta có thể gọi được chúng ra bên ngoài ?.

Bởi vì khi ta khởi tạo lớp ABC, lập tức sẽ gọi autoload(). Và phương thức autoload() sẽ lấy tên lớp ABC như 1 tham số \$url và dùng lệnh require() để nạp file tên ABC.php vào.

Khi đó, file ABC.php của chúng ta có lớp tên ABC và cũng có phương thức tên demo() nên file test.php của chúng ta mới xuất giá trị của phương thức demo() ra bên ngoài được.

Bạn thấy đấy, thay vì trước khi sử dụng 1 lớp ta phải require() các lớp ở trên đầu rất nhiều. Thì giờ đây với **lazy loading** ta chỉ việc gọi lớp thì tự động hệ thống sẽ nạp file ứng với tên lớp đó. Điều này giúp ta bớt nhiều thời gian và giảm hao tổn tài nguyên khi phải thực hiện việc nạp load các thư viện ở trên đầu các file rất nhiều.

Tuy nhiên, cần lưu ý rằng. Khi sử dụng cơ chế này, thường ta phải chú ý tới tên lớp và tên file. Chúng cần phải có sự liên quan nhất định. Ví dụ: Tên lớp và tên file giống nhau. Vì nếu không giống nhau thì quá trình nạp load sẽ thất bại. Và dĩ nhiên ta không thể lấy được các lớp, phương thức như mong muốn.

Kỹ thuật **lazy loading** cũng là kỹ thuật được sử dụng rất nhiều trong các [PHP Framework](#) hiện nay. Chẳng hạn như [zend framework](#), khi bạn muốn gọi 1 lớp thường thì sẽ gọi như sau:

PHP Code:

```
$model=new Zend_Db_Table_Abstract;
```

Có nghĩa là bạn đang truy cập vào thư mục Zend vào tiếp thư mục Db, vào tiếp thư mục Table và vào file Abstract.php. Zend/Db/Table/Abstract.php.

Và dĩ nhiên trong file Abstract.php phải có lớp tên là Zend_Db_Table_Abstract:

PHP Code:

```
class Zend_Db_Table_Abstract{  
    //code ở đây  
}
```

Để autoload hiểu điều đó ta có thể định nghĩa đơn giản như sau:

PHP Code:

```
function __autoload($url) {
    $url=str_replace("_","/",$url);
    require("$url.php");
}
```

Như bạn thấy, ta tìm kiếm tất cả các dấu gạch dưới trong lớp và thay thế chúng bằng dấu "/". Khi đó hệ thống sẽ nạp đúng như quy trình trên.

Regular Expressions

1. Regular expression là gì?

+ Biểu thức chính quy.

+ Hiểu nôm na là 1 chuỗi có quy tắc để mô tả những chuỗi(string) khác

2. Cú pháp cơ bản (ở đây mình chỉ trình bày và ví dụ cho ngôn ngữ php)

Example:

PHP Code:

```
<?php
$re = '/hello/'; // biểu thức chính quy cho một string có chuỗi "hello"
ở trong đó.
$str = 'hello world';
if(preg_match($re, $str)) {
    echo 'Yes';
}
?>
```

Output: Yes

+ Ký hiệu “^” và “\$”: bắt đầu và kết thúc 1 string

Example:

PHP Code:

```
<?php
$re = '/^hello/'; // biểu thức chính quy cho một string bắt đầu bởi ch
uỗi "hello"
$str = 'hello world';
if(preg_match($re, $str)) {
    echo 'Yes';
}
else {
    echo 'No';
}
?>
```

Output: Yes

PHP Code:

```
<?php
$re = '/hello$/'; // biểu thức chính quy cho một string kết thúc bởi ch
uỗi "hello"
$str = 'hello world';
if(preg_match($re, $str)) {
    echo 'Yes';
}
}
```

```

else {
    echo 'No';
}
?>

```

Output: No

+ Ký hiệu: “*”, “+”, “?”

\$re = '/^ab*\$/'; // biểu thức chính quy cho một string bắt đầu bởi a, và kết thúc là 0 hoặc nhiều b (ví dụ: a, ab, abb, abbb, ...);

\$re = '/^ab+\$/'; // biểu thức chính quy cho một string bắt đầu bởi a, và kết thúc là 1 hoặc nhiều b (ví dụ: ab, abb, abbb, ...);

\$re = '/^ab?\$/'; // biểu thức chính quy cho một string bắt đầu bởi a, và kết thúc là b hoặc là không (ví dụ: ab hoặc a).

Example:

PHP Code:

```

<?php
$re = '/^ab*$/';
$str = 'abbc';
if(preg_match($re, $str)) {
    echo 'Yes';
}
else {
    echo 'No';
}
?>

```

Output: No

+ Sử dụng: {}:

\$re = '/^ab{2}\$/'; // biểu thức chính quy cho một string bắt đầu bởi a, và kết thúc là 2 chữ b (là abb);

\$re = '/^ab{2,}\$/'; // biểu thức chính quy cho một string bắt đầu bởi a, và kết thúc là ít nhất 2 chữ b (ví dụ: abb, abbb, abbbb, ...);

\$re = '/^ab{2,5}\$/'; // biểu thức chính quy cho một string bắt đầu bởi a, và kết thúc là ít nhất 2 chữ b và nhiều nhất là 5 chữ b (ví dụ: abb, abbb, abbbb, abbbbb);

+ Sử dụng : () và |

\$re = '/^a(bc)*\$/'; // biểu thức chính quy cho một string bắt đầu bởi a, và kết thúc là 0 hoặc nhiều 'bc' (ví dụ abc, abcbc, abcbcbbcbc, ...)

\$re = '/^a(b|c)*\$/'; // biểu thức chính quy cho một string bắt đầu bởi a, và kết thúc là 0 hoặc nhiều 'b' hoặc nhiều 'c' hoặc 'b' 'c' lẫn lộn :D (ví dụ abc, abcccccccccc, abccccbbbc, ...)

+ Sử dụng symbol '!': đại diện cho một ký tự đơn bất kỳ

\$re = '/^.{3}\$/'; //Biểu thức chính quy cho một chuỗi có đúng 3 ký tự bất kỳ.

PHP Code:

```

<?php
$re = '/^.{3}$/';
//Biểu thức chính quy cho một chuỗi có đúng 3 ký tự bất kỳ.
$str = '&#%';
if(preg_match($re, $str)) {
    echo 'Yes';
}
else {
    echo 'No';
}

```

```
}  
?>
```

Output: Yes

+ **Sử dụng:** '-':

[0-9] : Một chữ số

[a-zA-Z]: một ký tự A->Z, a->z

[a-d] : ~ (a|b|c|d)

[^a-zA-Z]: một ký tự không phải là A->Z, a->z

[^0-9]: một ký tự không phải là số

+ **Sử dụng:** '\'

\d - Chữ số bất kỳ ~ [0-9]

\D - Ký tự bất kỳ không phải là chữ số (ngược với \d) ~ [^0-9]

\w - Ký tự từ a-z, A-Z, hoặc 0-9 ~ [a-zA-Z0-9]

\W - Ngược lại với \w (nghĩa là các ký tự không thuộc các khoảng: a-z, A-Z, hoặc 0-9)

~[^a-zA-Z0-9]

\s - Khoảng trắng (space)

\S - Ký tự bất kỳ không phải là khoảng trắng.

3. Các hàm cơ bản vận dụng regular expression

+ **preg_match** : <http://php.net/manual/en/function.preg-match.php>

```
int preg_match ( string $pattern , string $subject [, array &$matches [, int $flags = 0 [, int $offset = 0 ]]] )
```

Cơ bản là để tìm kiếm 1 string có làm việc theo một \$re.

Ví dụ:

PHP Code:

```
<?php  
$re = '/^\w+$/';  
// một string toàn ký tự A->Z, a->z, 0->9  
$str = 'quya*';  
if(preg_match($re, $str)) {  
    echo 'Yes';  
}  
else {  
    echo 'No';  
}  
?>
```

Output: No

+ **preg_replace**: <http://www.php.net/manual/en/function.preg-replace.php>

```
mixed preg_replace ( mixed $pattern , mixed $replacement , mixed $subject [, int $limit = -1 [, int &$count ] ] )
```

Cơ bản là để tìm kiếm trong 1 string những chuỗi có cấu trúc theo \$re để thay thế

Ví dụ:

PHP Code:

```
<?php  
$re = '/\w+$/';  
$str = '*quya';  
echo preg_replace($re, 'hi', $str);  
?>
```

Output: *hi

+ **preg_split**: <http://php.net/manual/en/function.preg-split.php>

array preg_split (string \$pattern , string \$subject [, int \$limit = -1 [, int \$flags = 0]])

Ví dụ:

Các bạn xem tạm ví dụ ở trong php manual nhé. Một rùi.

Giới thiệu Regular Expression Nâng cao

I. Thêm comment vào biểu thức Regular Expression:

Trong trường hợp biểu thức chính quy của bạn quá dài. Và muốn sau này người dùng sau có thể hiểu và chỉnh sửa lại (reuse), thì chúng ta cần phải ghi chú lại. hi hi.

Để thêm comment vào 1 biểu thức Regular Expression hiện giờ có 2 cách thì phải.

- Sử dụng cú pháp (?# nội dung ghi chú).

Ví dụ: kiểm tra ngày có đúng định dạng dd/mm/yyyy không ta có parttern sau:

Mã:

```
$parttern = "/^\d{2}\\/\d{2}\\/\d{4}$/" //cái này ngắn quá tham khảo thôi.
```

Sau khi thêm comment vào:

Mã:

```
$parttern = "/^\d{2}(?# ngày là số nguyên 2 kí tự)\\/\d{2}(?# Tháng là số nguyên 2 kí tự)\\/\d{4}(?# Năm là số nguyên 4 kí tự)$/"
```

- Cách này nhìn sáng sủa và dễ hiểu hơn:

Mã:

```
$parttern = "/^
    \d{2}    # ngày là số nguyên 2 kí tự
    \/      # dấu phân cách "/"
    \d{2}    # Tháng là số nguyên 2 kí tự
    \/      # dấu phân cách "/"
    \d{4}    # Năm là số nguyên 4 kí tự
$/x"; // Sử dụng modifier là x.
```

Khi sử dụng modifier x thì khoảng trắng trong biểu thức chính quy sẽ được bỏ qua, nếu không nó sẽ được escaped (\s). Điều này dễ dàng để thêm comment. Comment sẽ bắt đầu bằng dấu “#” và kết thúc tại 1 dòng mới.

II. Greedy và Ungreedy:

Để hiểu được phần này bạn cần xem qua ví dụ sau:

Ví dụ: Mình có 1 đoạn mã html sau khi view source của 1 trang web

Mã:

```
<html>
<head><title>Greedy và Ungreedy</title></head>
<body>
    <p id="weather">
        Nội dung cần lấy nằm ở đây
    </p>
    <p>
        Nội dung khác
```

```
</p>
</body>
</html>
```

Vậy để lấy nội dung trong thẻ `<p id="weather"></p>`, mình sẽ phải viết biểu thức sau:

PHP Code:

```
$parttern = "</p id=\"weather\">(.*?)</p>/s";
if(preg_match($parttern, $html, $match))
{
    print_r($match);
}
```

Hi hi, kết quả thế nào các bạn sẽ lấy được nội dung cần lấy chứ.

Mã:

```
Array (
[0] =>
Nội dung cần lấy ở đây
```

```
Nội dung khác
[1] =>
Nội dung cần lấy ở đây
Nội dung khác
)
```

Như vậy, nội dung mình cần lấy vẫn chưa lấy được, tìm hiểu nguyên nhân tại sao:

Xem lại \$parttern của mình: \$parttern = `</p id="weather">(.*?)</p>/s`;

Mô tả parttern: bắt đầu bằng thẻ p có id="weather", tiếp sau là nội dung bất kì có thể có nhiều hoặc không có kí tự nào và kết thúc là html tag `</p>`.

Khi dùng modifier s: cho biết so khớp parttern với mọi kí tự kể cả kí tự xuống dòng, nếu không có nó kí tự xuống dòng sẽ bị loại trừ.

- Mặc định trong so khớp mẫu thì nó sẽ lấy thẻ p có id = "weather" và nội dung bất kì bên trong đến khi gặp tag đóng `</p>`. Nhưng phải là tag đóng `</p>` cuối cùng thì nó mới dừng lại (Greedy, tiếng việt là tham lam, tham ăn gì đó) so tới cuối luôn đó mà.

- Vậy làm sao để parttern hiểu rằng khi so khớp mình chỉ lấy nội dung bất kì giữa 2 tag `<p id="weather">` nội dung đến tag đóng `</p>` đầu tiên thôi. Lúc đó, mình sẽ dùng kí tự dấu hỏi "?" (Ungreedy) để thêm vào sau [quantifier](#) của parttern muốn so khớp. Hi hi Cái quantifier tham khảo lại nha, vì bài này đang hướng dẫn Advanced Regular Expression mà.

- Vì vậy để lấy được nội dung chính xác với parttern của mình

Mã:

```
$parttern = '</p id="weather">(.*?)<\p>/s';
thì mình chỉ cần thêm dấu hỏi sau quantifier "*" là xong.
```

Mã:

```
$parttern = '</p id="weather">(.*?)<\p>/s';
```

Chạy lại code

PHP Code:

```
$parttern = "</p id=\"weather\">(.*?)</p>/s";
if(preg_match($parttern, $html, $match))
{
    print_r($match);
}
```

bạn sẽ nhận được kết quả như ý

Mã:

```
Array (
[0] =>
    Nội dung cần lấy ở đây
[1] =>
    Nội dung cần lấy ở đây
)
```

Đệ quy với menu không giới hạn số cấp (N cấp)

Mấy hôm nay lướt qua qhonline thấy nhiều bạn comment về menu đa cấp quá. cái này mình cũng từng làm nên mình viết 1 bài hy vọng giúp được một vài bạn.

Đặt vấn đề: Hầu hết các tut trên diễn đàn đều là menu hạn chế số cấp. vì thế mình sẽ hướng dẫn các bạn xử lý đệ quy với số cấp không giới hạn.

Giải pháp tối ưu khi xử lý đệ quy:

Giải pháp 1: đệ quy với câu truy vấn sql.

Với giải pháp này thì đoạn code chúng ta sẽ có quy trình như sau

- Mỗi lần đệ quy sẽ lấy menu từ database với điều kiện where id_parent trong db = id_parent truyền vào. như thế giả sử menu chúng ta có n cấp thì chúng ta sẽ phải select với n câu truy vấn.

Giải pháp 2: lấy hết menu ra một mảng rồi mỗi lần đệ quy sẽ duyệt qua cái mảng đó, không cần phải truy vấn lại

Cả 2 giải pháp đều ok cho kết quả như nhau, nhưng chúng ta cùng bàn lại xem giải pháp nào sẽ tốt hơn. với giải pháp 1 thì n câu truy vấn cho cái menu thì quá là tệ hại. còn giải pháp 2 ta chỉ mất 1 câu truy vấn và phần công việc còn lại là do code php xử lý. nên sẽ ít tốn tài nguyên hơn

Giải quyết:

Mình sẽ lựa chọn giải pháp 2 cho cái menu của mình.

Giả sử mình có một database với các thông số [id | name | link | id_parent].

Vì mình ko có database nên mình sẽ dùng mảng này thay thế coi như nó được lấy từ database

PHP Code:

```
$menus = array(
    array('id' => '1', 'name' => 'LEVEL 1', 'link' => 'localhost/trangc
hu.php', 'parent' => '0'),
    array('id' => '2', 'name' => 'LEVEL 2', 'link' => 'localhost/trangc
hu.php', 'parent' => '1'),
    array('id' => '3', 'name' => 'LEVEL 3', 'link' => 'localhost/trangc
hu.php', 'parent' => '2'),
    array('id' => '4', 'name' => 'LEVEL 4', 'link' => 'localhost/trangc
```

```

hu.php', 'parent' => '3'),
    array('id' => '5', 'name' => 'LEVEL 1', 'link' => 'localhost/trangc
hu.php', 'parent' => '0'),
    array('id' => '6', 'name' => 'LEVEL 1', 'link' => 'localhost/trangc
hu.php', 'parent' => '0'),
    array('id' => '7', 'name' => 'LEVEL 2', 'link' => 'localhost/trangc
hu.php', 'parent' => '6'),
    array('id' => '8', 'name' => 'LEVEL 2', 'link' => 'localhost/trangc
hu.php', 'parent' => '6'),
    array('id' => '9', 'name' => 'LEVEL 2', 'link' => 'localhost/trangc
hu.php', 'parent' => '6'),
    array('id' => '10', 'name' => 'LEVEL 3', 'link' => 'localhost/trang
chu.php', 'parent' => '9'),
    array('id' => '11', 'name' => 'LEVEL 3', 'link' => 'localhost/trang
chu.php', 'parent' => '9'),
    array('id' => '12', 'name' => 'LEVEL 3', 'link' => 'localhost/trang
chu.php', 'parent' => '9')
);

```

Code show menu đệ quy

PHP Code:

```

function show_menu($menus = array(), $parent = 0)
{
    // LAY MENU DUA VAO MENU ID MENU CHA TRUYEN VAO
    $current_menus = array();
    foreach ($menus as $key => $val) {
        if ($val['parent'] == $parent) {
            $current_menus[] = $val;
            unset($menus[$key]);
        }
    }

    // SHOW RA MENU THEO UL
    if (sizeof($current_menus) > 0) {
        echo '<ul>';
        foreach ($current_menus as $key => $val) {
            echo '<li><a href="' . $val['link'] . '">', $val['name'], '</
a>';

            show_menu($menus, $val['id'], false);
            echo '</li>';
        }
        echo '</ul>';
    }
}

```

ý tưởng như sau. sau khi có cái mảng đó rồi thì mình dùng mã php để select theo điều kiện thay cho sql. biến \$current_menus chính là menu cần hiển thị hiện tại

Và cuối cùng là show menu ra thôi

PHP Code:

```
show_menu($menus);
```

Code full:

PHP Code:

```

$menus = array(
    array('id' => '1', 'name' => 'LEVEL 1', 'link' => 'localhost/trangc
hu.php', 'parent' => '0'),
    array('id' => '2', 'name' => 'LEVEL 2', 'link' => 'localhost/trangc
hu.php', 'parent' => '1'),
    array('id' => '3', 'name' => 'LEVEL 3', 'link' => 'localhost/trangc
hu.php', 'parent' => '2'),
    array('id' => '4', 'name' => 'LEVEL 4', 'link' => 'localhost/trangc
hu.php', 'parent' => '3'),
    array('id' => '5', 'name' => 'LEVEL 1', 'link' => 'localhost/trangc
hu.php', 'parent' => '0'),
    array('id' => '6', 'name' => 'LEVEL 1', 'link' => 'localhost/trangc
hu.php', 'parent' => '0'),
    array('id' => '7', 'name' => 'LEVEL 2', 'link' => 'localhost/trangc
hu.php', 'parent' => '6'),
    array('id' => '8', 'name' => 'LEVEL 2', 'link' => 'localhost/trangc
hu.php', 'parent' => '6'),
    array('id' => '9', 'name' => 'LEVEL 2', 'link' => 'localhost/trangc
hu.php', 'parent' => '6'),
    array('id' => '10', 'name' => 'LEVEL 3', 'link' => 'localhost/trang
chu.php', 'parent' => '9'),
    array('id' => '11', 'name' => 'LEVEL 3', 'link' => 'localhost/trang
chu.php', 'parent' => '9'),
    array('id' => '12', 'name' => 'LEVEL 3', 'link' => 'localhost/trang
chu.php', 'parent' => '9')
);

function show_menu($menus = array(), $parent = 0)
{
    // LAY MENU DUA VAO ID MENU CHA
    $current_menus = array();
    foreach ($menus as $key => $val) {
        if ($val['parent'] == $parent) {
            $current_menus[] = $val;
            unset($menus[$key]);
        }
    }

    // SHOW RA MENU THEO UL
    if (sizeof($current_menus) > 0) {
        echo '<ul>';
        foreach ($current_menus as $key => $val) {
            echo '<li><a href="' . $val['link'] . '">', $val['name'], '</
a>';

            show_menu($menus, $val['id'], false);
            echo '</li>';
        }
        echo '</ul>';
    }
}

show_menu($menus);

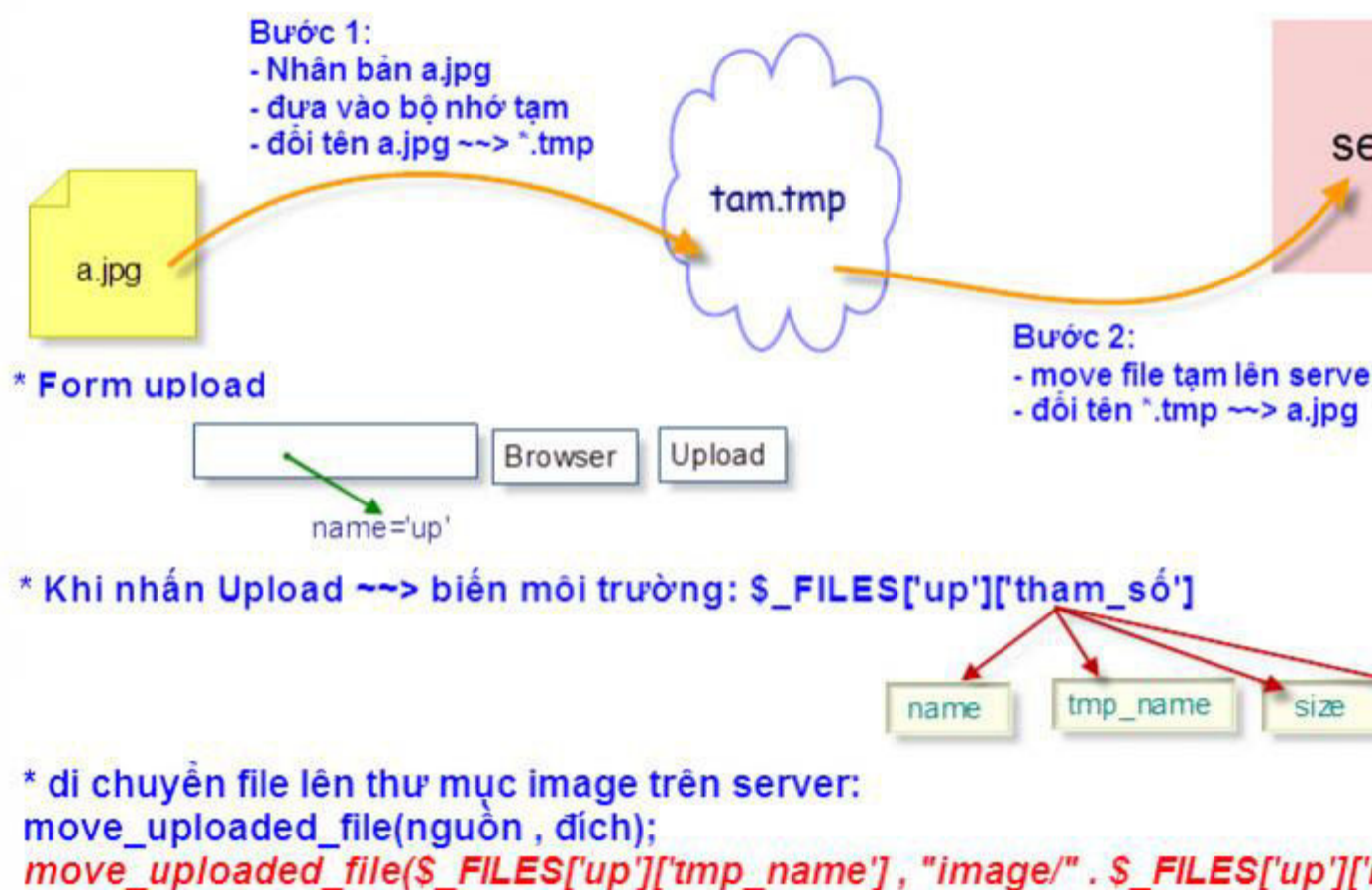
```

Các bạn copy đưa vào file PHP bình thường sẽ thấy thành quả :D

Upload file trong php

1/ Cơ chế upload file trong php

UPLOAD FILE TRONG PHP GỒM 2 BƯỚC



2/ VD upload file + form upload

FORM UPLOAD

```
1 <html>
2 <head>
3     <title>upload file</title>
4 </head>
5 <body>
6 <form action='xulyupload.php' method='post' name='fupload' enctype='multipart/form-data'>
7     Chọn file <input type='file' name='up' size='25' />
8     <input type='submit' name='sub' value='Upload' />
9 </form>
10 </body>
11 </html>
```

+ Xử lý upload

XỬ LÝ UPLOAD

```
1 <?php
2 if($_FILES['up']['name']!=NULL){
3     move_uploaded_file($_FILES['up']['tmp_name'], "upload/".$_FILES['up']['name']);
4     echo "<h3>Upload thành công</h3>";
5     $name=$_FILES['up']['name'];
6     $tmp=$_FILES['up']['tmp_name'];
7     $size=$_FILES['up']['size'];
8     $type=$_FILES['up']['type'];
9     echo "<h3>Tên gốc của hình là: $name</h3>";
10    echo "<h3>Tên tạm của hình là: $tmp</h3>";
11    echo "<h3>Kích thước của hình tính theo byte là: $size</h3>";
12    echo "<h3>Định dạng của hình là: $type</h3>";
13    echo "<img src='upload/$name' />";
14 }else{
15     echo "Vui lòng chọn file upload";
16 }
17 ?>
```

Class support Mutil Language cho cơ bản và nâng cao và cho cả các framework khác!!!!

Mình thấy các bạn post bài cần mutil language trên website... vì vậy mà mình viết class này cho các bạn.

B1: Bạn tạo file **index.php** có nội dung như sau

PHP Code:

```
<?php

session_start();
ob_start();
/* * * error reporting on * * */
error_reporting(E_ALL);

/* * * define the path * * */
$site_path = realpath(dirname(__FILE__));
define('__SITE_PATH', $site_path . '/');
define('__LANG_PATH', $site_path . '/language/');
define('__EXT', '.php');

include __SITE_PATH . 'classes/' . 'language.class.php';

Language::locate_set('en');// SET LA ENGLISH HAY VIETNAM
//VI DU
//ENGLISH = EN THÌ TẠO THƯ MỤC EN TƯƠNG ỨNG TRONG THƯ MỤC LANGUAGE
//VIETNAM = VN THÌ TẠO THƯ MỤC VN TƯƠNG ỨNG TRONG THƯ MỤC LANGUAGE

echo Language::lang('home_lang.home_title');

?>
```

B2: Tạo thư mục **classes** đồng cấp với file **index.php**, và tạo 1 file **language.class.php** trong thư mục **classes** với nội dung như sau

PHP Code:

```
<?php

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

Class Language {

    protected static $locale;
    protected static $language_cache = array();

    public static function locate_set($key = 'vn') {
        self::$locale = $key;
    }

    public static function find_file($filename) {
        $ext = '.php';
        //Search Path
        $search = __LANG_PATH . self::$locale . '/' . $filename . __EXT
;

        $found = false;
        if (is_file($search)) {
            // A matching file has been found

```


B5: Tạo file **home_lang.php** trong thư mục **en** với nội dung như sau

PHP Code:

```
<?php

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

$lang['home_title'] = 'Home';
?>
Sử dụng 1 cách dễ dàng
PHP Code:
<?php echo Language::lang('home_lang.home_title'); ?>
Thay đổi ngôn ngữ các bạn set lại locate cho nó bằng cách
PHP Code:
<?php Language::locate_set('en'); ?>
```

Đây chỉ là đa ngôn ngữ phần giao diện

Hướng dẫn CSS Framework 960 cho mọi người

Tham gia VC và QHO lâu rồi nhưng chưa đóng góp được gì nhiều, hôm nay mạng phép viết 1 bài hy vọng góp chút kiến thức cho các bạn. Lần đầu tiên viết tutorial mong mọi















người ủng hộ. Yêu cầu kiến thức cơ bản về css, ta vào vấn đề



Có ai đã từng sử dụng 1 framework css nào chưa và 960css framework có gì. Có lẽ các bạn đều cho rằng css khá đơn giản nên cũng ko nhất thiết phải tìm hiểu sâu đến framework làm gì. 960 css framework sẽ cho bạn 1 cái nhìn khác về điều này

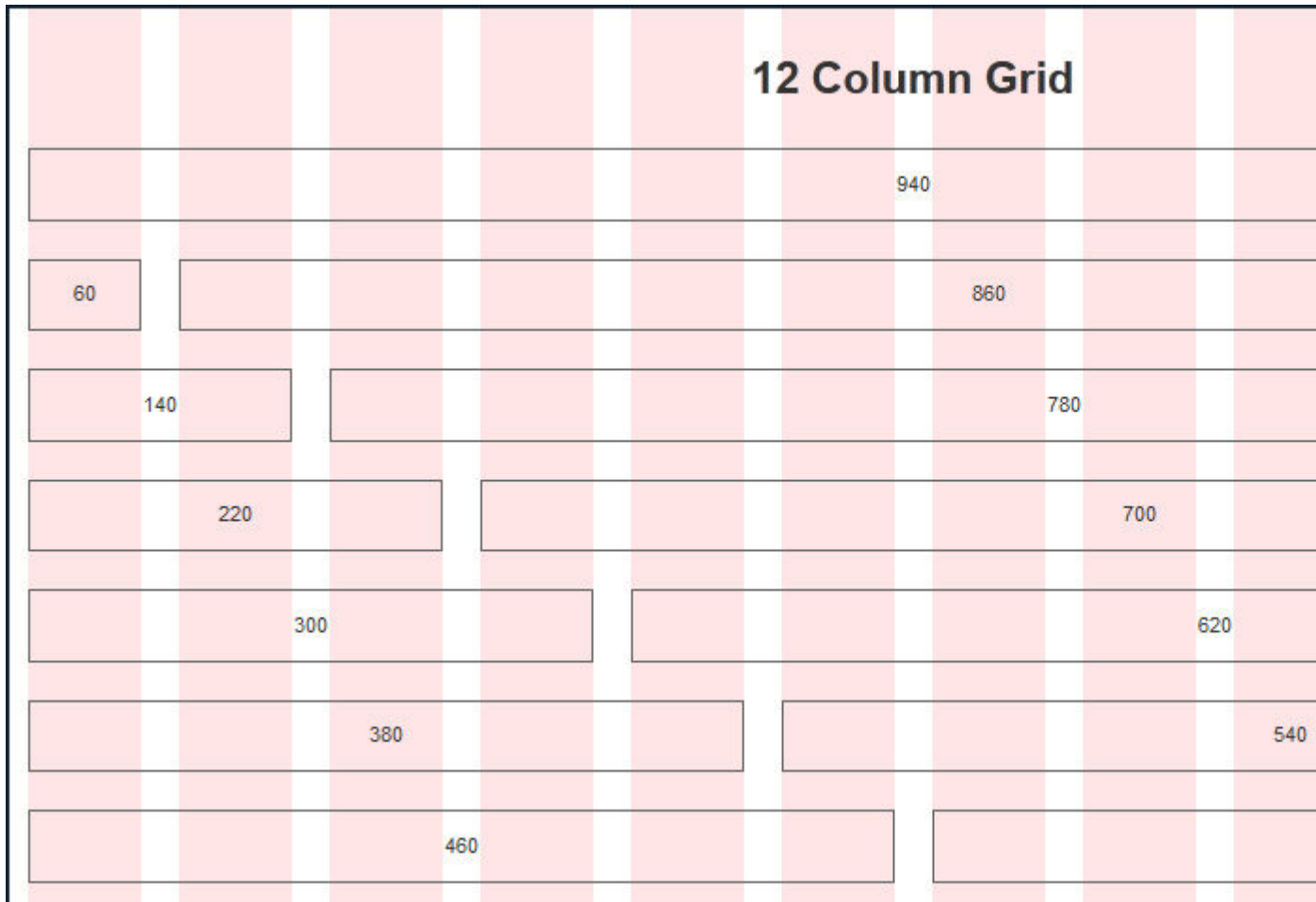
Đầu tiên download framework <https://nodeload.github.com/nathansm...zipball/master>

Giải nén -> mở thư mục code -> css ta được

 min	3/2/2012 7:08 AM	File folder	
 960.css	27/2/2012 10:09 AM	Cascading Style Sh...	10 KB
 960_12_col.css	3/2/2012 7:08 AM	Cascading Style Sh...	6 KB
 960_12_col_rtl.css	3/2/2012 7:08 AM	Cascading Style Sh...	6 KB
 960_16_col.css	3/2/2012 7:08 AM	Cascading Style Sh...	7 KB
 960_16_col_rtl.css	3/2/2012 7:08 AM	Cascading Style Sh...	7 KB
 960_24_col.css	3/2/2012 7:08 AM	Cascading Style Sh...	9 KB
 960_24_col_rtl.css	3/2/2012 7:08 AM	Cascading Style Sh...	9 KB
 960_rtl.css	3/2/2012 7:08 AM	Cascading Style Sh...	10 KB
 demo.css	27/2/2012 11:02 AM	Cascading Style Sh...	1 KB
 reset.css	3/2/2012 7:08 AM	Cascading Style Sh...	3 KB
 reset_rtl.css	3/2/2012 7:08 AM	Cascading Style Sh...	3 KB
 text.css	3/2/2012 7:08 AM	Cascading Style Sh...	2 KB
 text_rtl.css	3/2/2012 7:08 AM	Cascading Style Sh...	2 KB

Chỉ cần sử dụng copy 960.css để sử dụng. Bạn có thể vào min để lấy file dung lượng nhỏ hơn để up lên host.

Cái tên 960 bắt nguồn từ việc framework này cung cấp cho chúng ta 1 mô hình website được chia làm nhiều cột. Với độ rộng của site là 960px được chia làm 12 cột, 16 cột hay 24 cột <http://960.gs/demo.html>



960px là độ rộng phù hợp với hầu hết các website hiện nay. Margin mỗi cột là 20px tạo cái nhìn thoáng cho website.

Container & Grid

- Container để phân biệt số cột cho các thành phần bên trong như 12, 16, 24
- Grid xác định số cột

VD bạn muốn tạo 1 khối kích thước = 3 cột:

Lưu ý: phải có khối `<div class="container_12">` bao bên ngoài

HTML Code:

```
<div class="container_12">
  <div class="grid_3" style="background:#666;"> 3 Cột</div>
</div>
```



Nhiều cột cùng lúc

HTML Code:

```
<div class="container_12">
```

```

    <div class="grid_3" style="background:blue;">First</div>
    <div class="grid_5" style="background:red;">Second</div>
    <div class="grid_4" style="background:green;">Third</div>
</div>

```



Pull & Push

- Đẩy khối qua trái - phải tính theo cột

VD: ta đẩy 2 khối First qua phải 1 ô và Second qua trái 2 ô

HTML Code:

```

<div class="container_12">
    <div class="grid_3 push_1" style="background:blue;">3 Cột</div>
    <div class="grid_5" style="background:red;">5 Cột</div>
    <div class="grid_4 pull_2" style="background:green;">4 Cột</div>
</div>

```



Suffix & Prefix

- Tăng padding phải - trái, tính theo cột

VD: Ta muốn tăng padding-right khối First 1 cột vào padding-left khối Second 1 cột

HTML Code:

```

<div class="container_12">
    <div class="grid_3 suffix_1" style="background:blue;">3 Cột3
    Cột3 Cột3 Cột3 Cột3 Cột3 Cột3 Cột3 Cột3 Cột3 Cột3 Cột3 Cột3 Cột3 Cột3 Cột3
    Cột</div>
    <div class="grid_5 prefix_1" style="background:red;">5 Cột</div>
    <div class="grid_4" style="background:green;">4 Cột</div>
</div>

```



Vì tăng padding nên độ rộng các khối cũng tăng theo. Tuy nhiên nội dung của khối First vẫn ko tràn qua cột thứ 4 mà vẫn chiếm 3 cột

Alpha & Omega

- Xóa margin trái - phải khối class GRID

Ta xem VD sau:

HTML Code:

```

<div class="container_12">
<div class="grid_12">

```

```

    <div class="grid_3" style="background:blue;">3 Cột</div>
    <div class="grid_5" style="background:red;">5 Cột</div>
    <div class="grid_4" style="background:green;">4 Cột</div>
  </div>
</div>

```



Tại sao khối Frist lại dịch qua phải 10px mà ko nằm đúng vị trí. Nguyên nhân là nó đang nằm trong 1 khối Grid khác. Và thuộc tính margin của nó khiến nó phải dịch qua phải 10px. Ta khắc phục = cách sử dụng class "alpha" và "omega"

HTML Code:

```

<div class="container_12">
<div class="grid_12">
    <div class="grid_3 alpha" style="background:blue;">3 Cột</div>
    <div class="grid_5" style="background:red;">5 Cột</div>
    <div class="grid_4 omega" style="background:green;">4 Cột</div>
  </div>
</div>

```



Đó là tất cả những gì mình biết về 960 css framework.

Điểm mạnh:

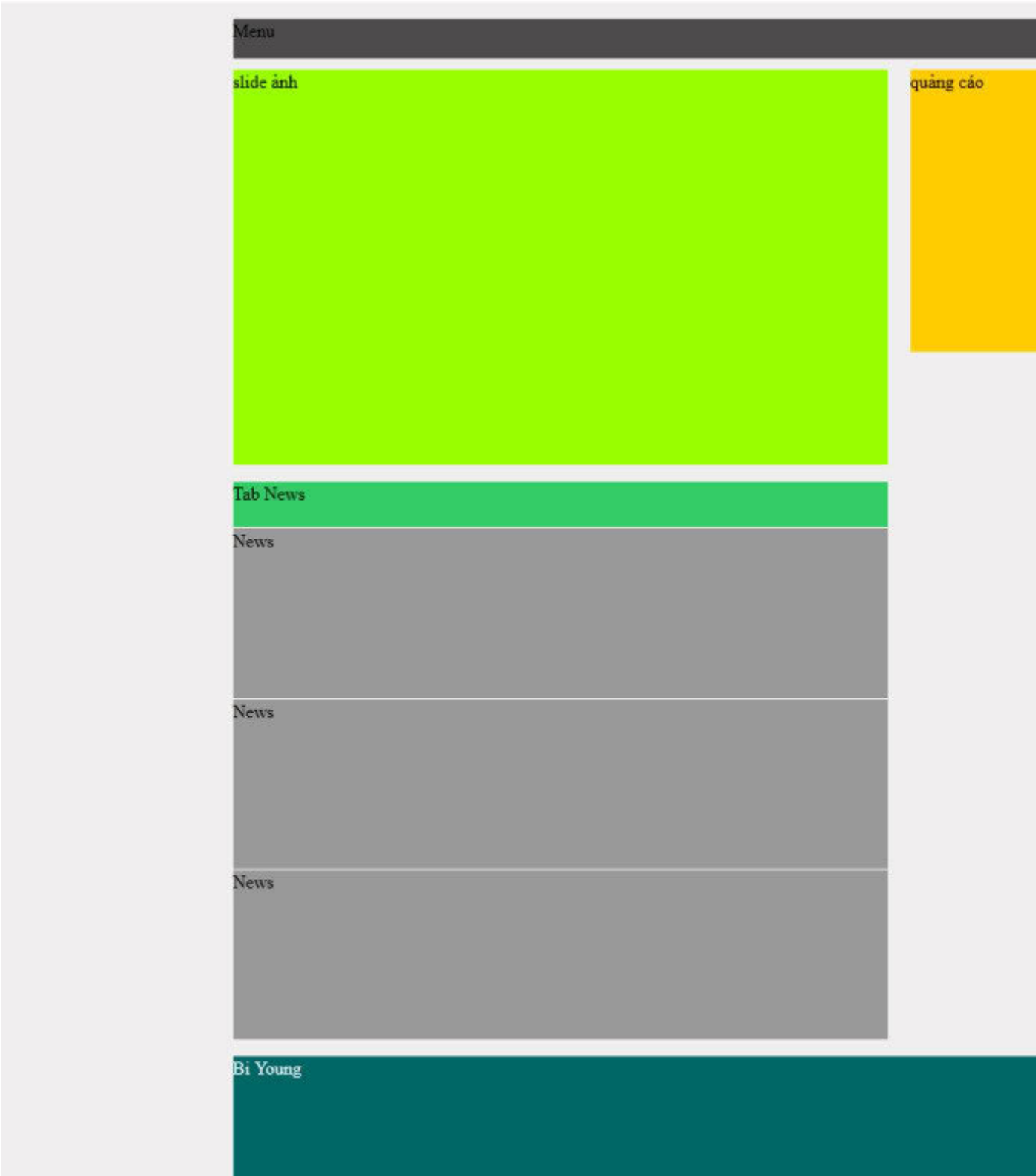
- Tiết kiệm thời gian
- Tương thích nhiều trình duyệt
- Thiết kế theo chuẩn

Khuyết:

- Độ rộng site tối đa là 960px
- Phụ thuộc vào framework

Với những gì 960 css framework mang lại việc tạo nên các layout cho website trở nên rất nhanh chóng. Hãy thử tạo layout đơn giản cho trang Genk.vn nào các bạn, mình mất chưa

đến 5p nhé 🐼 <http://i1254.photobucket.com/albums/...mfnamoka/8.jpg>



1 số website được tạo từ 960 css framework:

- <http://keynotekungfu.com/?ref=960>
- <http://brandrichmedia.com/>
- <http://nickfinck.com/>
- <http://drupal.org/>
- <http://5by5.tv/>

Nhập dữ liệu từ file excel sử dụng công nghệ XML...

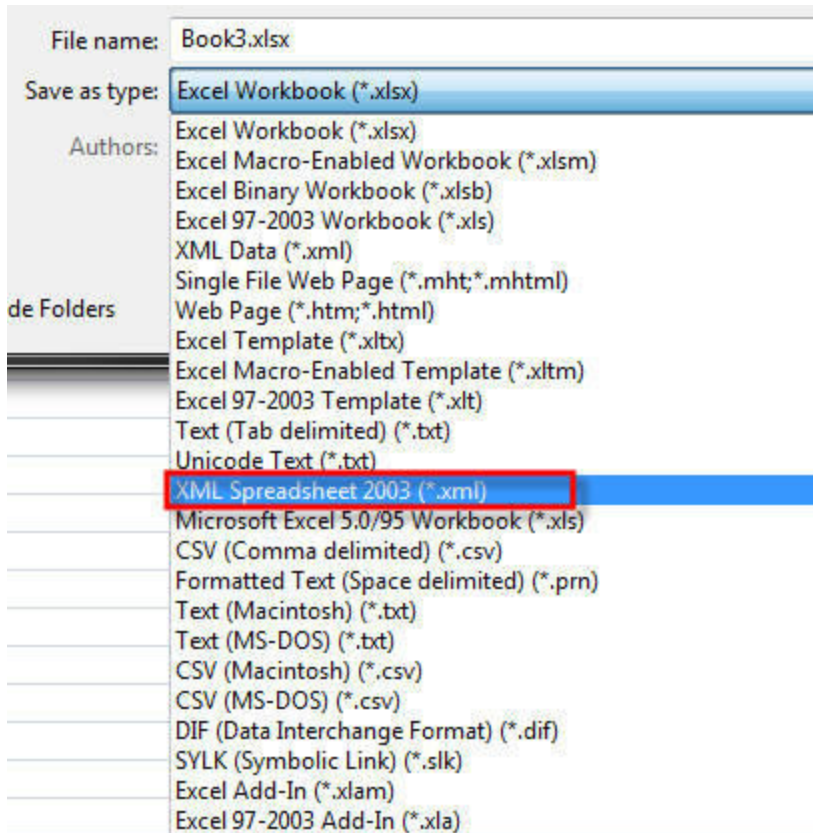
Nếu chúng ta thêm 1 user, 1 thể loại thì dễ dàng, còn nếu yêu cầu chúng ta hằng ngày nhập 1 list danh sách(>500) các thông tin của nhân viên, thông tin bài nhạc... để up lên web thì sao, chả lẽ chúng ta phải nhập bằng tay từng thông tin 1 hay sao, ngày nào cũng vậy chắc đứt bóng sớm :)). Vậy là bắt đầu mình suy nghĩ, làm cách nào để up 1 phút là xong hết không nhỉ, giả dụ nếu mình có 1 file excel có đầy đủ hết thông tin của tất cả các bài hát(>1000 bài đi cho nó oách :D) nếu mà up thẳng 1 lèo vào database được thì đỡ quá nhỉ, không biết php làm được không ta? thầy thì chỉ mới dạy tới phần xml thoai, nên đối với mình thì pò..code :(. Thế là tới nhà bác Google hỏi thăm coi thế nào, lang thang vài vòng lượn zô 1 trang toàn... tiếng anh, khiếp thiết, mình nhìn nó, nó nhìn mình cười chả ai biết ai =)) may thay thấy 1 dòng đập vào mắt bảo convert file .xls sang file xml, nghĩ một hồi cái tự nhiên nảy ra ý tưởng và lao đầu vào "cột" ... ặc nhâm, lao đầu vào code :)).

hihi nói nhảm này giờ đủ rồi giờ chúng ta bắt đầu nhé.

Đầu tiên chúng ta có 1 file excel với dữ liệu về thông tin bài nhạc như thế này:

	A	B	C	D	
1	Tên bài hát	Ca sĩ	Nhạc sĩ	Thể loại	
2	1 2 3 chia đôi nỗi về	Quang Hà	Bảo Chính	Nhạc Trẻ	Album
3	1 2 3 ngôi sao	Sỹ Luân	Sỹ Luân	Nhạc Trẻ	Album
4	1000 lý do anh đặt ra	Sơn ca	Vũ Quốc Bình	Nhạc Trẻ	Album
5	A time for us	Ngân Khánh - Ngô Kiến Huy	LV Phạm Duy	Nhạc Hot	Album
6	Âm nhạc của tôi	Takej Minh Huy	Young Uno	Nhạc Hot	Album
7	Âm nhạc của tôi	Takej Minh Huy	Young Uno	Nhạc Hot	Album
8	Anh chỉ là hình bóng của người	Sỹ Luân	Sỹ Luân	Nhạc Hot	Album
9	Anh đi tìm em	Takej Minh Huy	Phúc Bô	Nhạc Hot	Album
10	Anh không hiểu	Quang Hà	Hồ Anh Dũng	Nhạc Hot	Album
11	Anh mê vợ bé	Sơn ca - Huy Vũ	Vũ Quốc Bình	Nhạc Hot	Album
12	Anh muốn thầm nói với em rằng	Quang Vinh	n/a	Nhạc Hot	Album
13	Anh nhớ ai	Sơn ca	Nguyễn Văn Mười Tiêu	Nhạc Hot	Album
14	Ảnh sao băng	Sỹ Luân	Sỹ Luân	Nhạc Hot	Album
15	Anh tin mình đã cho nhau một	Thu Thủy - Lương Bằng Quang	Lương Bằng Quang	Nhạc Hot	Album
16	Ảnh trắng khuya	Quang Hà	Lương Bằng Quang	Nhạc Hot	Album
17	Anh về	Quách An An		Nhạc Hot	Album
18	Anh xin lỗi	Lâm Chấn Hải	Cao Đình Trung	Nhạc Trẻ	Album
19	Ba ngọn nến lung linh	Hoàng Ngân-Xuân Nghi	Ngọc Lễ	Nhạc Thiếu Nhi	Album
20	Baby love	Anh Tài	Hồ Minh Phúc	Nhạc Thiếu Nhi	Album

Sau đó mở lên và lưu lại với đuôi là .xml



Và sau đó mở file xml này lên, nó sẽ có cấu trúc như thế này:

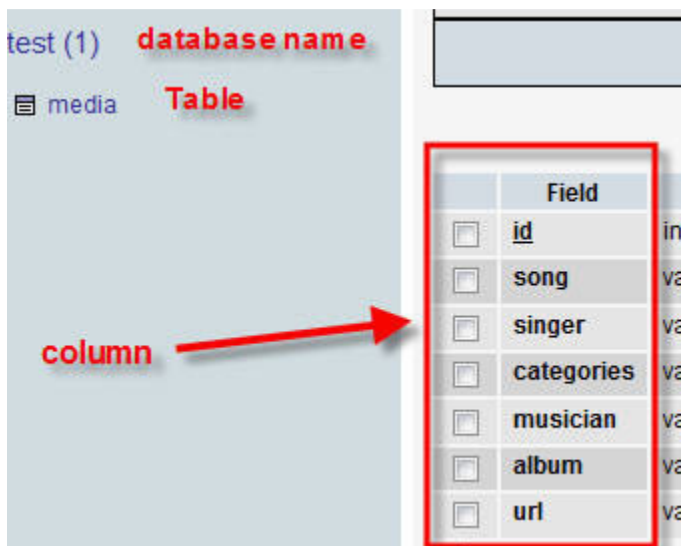
```

148 <Worksheet ss:Name="Sheet1">
149 <Table ss:ExpandedColumnCount="6" ss:ExpandedRowCount="20" x:FullColumns="1"
150 x:FullRows="1" ss:DefaultColumnWidth="54" ss:DefaultRowHeight="14.25">
151 <Column ss:AutoFitWidth="0" ss:Width="133.5"/>
152 <Column ss:Width="150"/>
153 <Column ss:Width="113.25"/>
154 <Column ss:Width="84.75"/>
155 <Column ss:Width="121.5"/>
156 <Column ss:AutoFitWidth="0" ss:Width="158.25"/>
157 <Row ss:Height="19.5">
158 <Cell ss:StyleID="s63"><Data ss:Type="String">Tên bài hát</Data></Cell>
159 <Cell ss:StyleID="s64"><Data ss:Type="String">Ca sĩ</Data></Cell>
160 <Cell ss:StyleID="s64"><Data ss:Type="String">Nhạc sĩ</Data></Cell>
161 <Cell ss:StyleID="s64"><Data ss:Type="String">Thể loại</Data></Cell>
162 <Cell ss:StyleID="s64"><Data ss:Type="String">Album</Data></Cell>
163 <Cell ss:StyleID="s64"><Data ss:Type="String">Tên tập tin</Data></Cell>
164 </Row>
165 <Row ss:Height="15.75">
166 <Cell ss:StyleID="s65"><Data ss:Type="String" x:Ticked="1">1 2 3 chia đôi lỗi v</Data></Cell>
167 <Cell ss:StyleID="s65"><Data ss:Type="String">Quang Hà</Data></Cell>
168 <Cell ss:StyleID="s65"><Data ss:Type="String">Bảo Chinh</Data></Cell>
169 <Cell ss:StyleID="s62"><Data ss:Type="String">Nhạc Trẻ</Data></Cell>
170 <Cell ss:StyleID="s67"><Data ss:Type="String">Album Miss you</Data></Cell>
171 <Cell ss:StyleID="s68"><Data ss:Type="String">1-2-3-chia-doi-loi-ve</Data></Cell>
172 </Row>
173 <Row ss:Height="15.75">
174 <Cell ss:StyleID="s65"><Data ss:Type="String">1 2 3 ngôi sao</Data></Cell>
175 <Cell ss:StyleID="s65"><Data ss:Type="String">Sỹ Luân</Data></Cell>
176 <Cell ss:StyleID="s65"><Data ss:Type="String">Sỹ Luân</Data></Cell>
177 <Cell ss:StyleID="s62"><Data ss:Type="String">Nhạc Trẻ</Data></Cell>
178 <Cell ss:StyleID="s67"><Data ss:Type="String">Album 1 2 3 ngôi sao</Data></Cell>
179 <Cell ss:StyleID="s68"><Data ss:Type="String">1-2-3-ngoi-sao</Data></Cell>
180 </Row>
181 <Row ss:Height="15.75">
182 <Cell ss:StyleID="s69"><Data ss:Type="String">1000 lý do anh đặt ra </Data></Cell>
183 <Cell ss:StyleID="s69"><Data ss:Type="String">Son ca</Data></Cell>
184 <Cell ss:StyleID="s69"><Data ss:Type="String">Vũ Quốc Bình</Data></Cell>
185 <Cell ss:StyleID="s62"><Data ss:Type="String">Nhạc Trẻ</Data></Cell>
186 <Cell ss:StyleID="s67"><Data ss:Type="String">Album 1 2 3 ngôi sao</Data></Cell>
187 <Cell ss:StyleID="s70"><Data ss:Type="String">1000-ly-do-anh-dat-ra</Data></Cell>
188 </Row>
189 <Row ss:Height="15">

```

Ở đây chúng ta thấy, Dữ liệu của chúng ta cần lấy nó nằm trong cặp thẻ <Row>, vậy chúng ta cần xác định được vị trí của thẻ <row> và sau đó dùng vòng lặp, lặp qua các thẻ <row> để lấy giá trị chúng ta cần sau đó nhập vào database, cụ thể chúng ta sao làm như sau:

Đầu tiên tạo 1 database tên test, sau đó tạo 1 bảng tên media với cấu trúc như sau:



Trong trang html ta có 1 form cho phép chọn file xml(convert từ file excel) cần nhập dữ liệu:

HTML Code:

```
<form action="b.php" method="post" enctype="multipart/form-data">
    Select a file: <input type="file" name="upload" size="50" />
    <input type="submit" name="send" value="Upload" />
</form>
```

Khi chọn file và nhấp vào nút upload code xử lý sẽ như sau:

PHP Code:

```
<?php

    if(isset($_POST["send"])){//Nếu nút send được nhấn
        if($_FILES["upload"]["name"] != NULL){//Nếu có chọn file
            move_uploaded_file($_FILES["upload"]["tmp_name"],"data/" .
$_FILES["upload"]["name"]);di chuyển file xml lên server trong thư mục
data

                $dom = new DOMDocument();//Khởi tạo đối tượng PHP DOM
                $dom->
>load("data/" . $_FILES["upload"]["name"]);//Load file xml vừa được chu
yển trong thư mục data
                $row = $dom->
>getElementsByTagName("Row");//Xác định vị trí của thẻ <Row>
                $first_row = TRUE;//Biến định là dòng đầu tiên trong bảng e
xcel(dòng tiêu đề)
                foreach($row as $r){//Bắt đầu lặp qua các thẻ Row
                    if(!$first_row){//Nếu không phải là dòng đầu tiên
//Lấy giá trị của các cột trên từng dòng
                        $a = $r->getElementsByTagName("Cell")->item(0)-
>nodeValue;
                        $b = $r->getElementsByTagName("Cell")->item(1)-
>nodeValue;
                        $c = $r->getElementsByTagName("Cell")->item(2)-
>nodeValue;
                        $d = $r->getElementsByTagName("Cell")->item(3)-
```

```

>nodeValue;
                                $e = $r->getElementsByName("Cell")->item(4)-
>nodeValue;
                                $f = $r->getElementsByName("Cell")->item(5)-
>nodeValue;
//Khởi tạo câu truy vấn
                                $sql = "insert into media(song,singer,musician,cate
gories,album,url) values('" . $a . "','" . $b . "','" . $c . "','" . $d
. "','" . $e . "','" . $f . "')";
//Thực thi câu truy vấn và kiểm tra kết quả trả về
                                if(mysql_query($sql) != FALSE){
                                    $mess ="Thêm dữ liệu thành công";
                                }else{
                                    $mess = "Thêm dữ liệu thất bại";
                                }
                                }
                                $first_row = FALSE;//Biến xác định đã qua dòng đầu tiên
                                }
                                echo "<cente>$mess</center>";
                            }
                        }
?>

```

Vậy là xong phần xử lý, bây giờ mình đổ dữ liệu ra ngoài bảng nhé

PHP Code:

```

<table cellpadding="0" cellspacing="0" border="1" width="80%">
    <tr>
        <th>STT</th>
        <th>Tên bài hát</th>
        <th>Ca sĩ</th>
        <th>Nhạc sĩ</th>
        <th>Thể loại</th>
        <th>Album</th>
        <th>Đường dẫn</th>
    </tr>
    <?php
        $sql = "select * from media order by categories ASC";
        $query = mysql_query($sql);
        if(mysql_num_rows($query) > 0){
            $stt = 0;
            while($data = mysql_fetch_assoc($query)){
                $stt++;
                echo "<tr>";
                echo "<td>$stt</td>";
                echo "<td>$data[song]</td>";
                echo "<td>$data[singer]</td>";
                echo "<td>$data[musician]</td>";
                echo "<td>$data[categories]</td>";
                echo "<td>$data[album]</td>";
                echo "<td>$data[url]</td>";
                echo "</tr>";
            }
        }else{
            echo "<tr><td colspan=7 align=center>Data empty</td></tr>";
        }
    </?php>

```

```
?>
</table>
```

Code đầy đủ cho toàn bộ quá trình của chúng ta:

HTML Code:

```
<?php
    $conn = mysql_connect("localhost","root","root");
    mysql_select_db("test",$conn);

    if(isset($_POST["send"])){
        if($_FILES["upload"]["name"] != NULL){
            move_uploaded_file($_FILES["upload"]["tmp_name"],"data/" .
$_FILES["upload"]["name"]);
            $dom = new DOMDocument();
            $dom->load("data/" . $_FILES["upload"]["name"]);
            $row = $dom->getElementsByTagName("Row");
            $first_row = TRUE;
            foreach($row as $r){
                if(!$first_row){
                    $a = $r->getElementsByTagName("Cell")->item(0)-
>nodeValue;
                    $b = $r->getElementsByTagName("Cell")->item(1)-
>nodeValue;
                    $c = $r->getElementsByTagName("Cell")->item(2)-
>nodeValue;
                    $d = $r->getElementsByTagName("Cell")->item(3)-
>nodeValue;
                    $e = $r->getElementsByTagName("Cell")->item(4)-
>nodeValue;
                    $f = $r->getElementsByTagName("Cell")->item(5)-
>nodeValue;

                    $sql = "insert into
media(song,singer,musician,categories,album,url) values('" . $a . "','"
. $b . "','" . $c . "','" . $d . "','" . $e . "','" . $f . "')";
                    if(mysql_query($sql) != FALSE){
                        $mess = "Them du lieu thanh cong";
                    }else{
                        $mess = "Them du lieu that bai" .
mysql_error();
                    }
                }
                $first_row = FALSE;
            }
            echo "<cente>$mess</center>";
        }
    }

?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>
    <meta http-equiv="content-type" content="text/html; charset=utf-
8" />
    <meta name="author" content="Le Kim Thuc" />
```

```

<style type="text/css">
    form{
        text-align: center;
        margin: 50px auto;
    }
</style>
</head>
<body>
    <form action="b.php" method="post" enctype="multipart/form-data">
        Select a file: <input type="file" name="upload" size="50" />
        <input type="submit" name="send" value="Upload" />
    </form>
    <table cellpadding="0" cellspacing="0" border="1" width="80%">
        <tr>
            <th>STT</th>
            <th>Tên bài hát</th>
            <th>Ca sĩ</th>
            <th>Nhạc sĩ</th>
            <th>Thể loại</th>
            <th>Album</th>
            <th>Đường dẫn</th>
        </tr>
        <?php
            $sql = "select * from media order by categories ASC";
            $query = mysql_query($sql);
            if(mysql_num_rows($query) > 0){
                $stt = 0;
                while($data = mysql_fetch_assoc($query)){
                    $stt++;
                    echo "<tr>";
                    echo "<td>$stt</td>";
                    echo "<td>$data[song]</td>";
                    echo "<td>$data[singer]</td>";
                    echo "<td>$data[musician]</td>";
                    echo "<td>$data[categories]</td>";
                    echo "<td>$data[album]</td>";
                    echo "<td>$data[url]</td>";
                    echo "</tr>";
                }
            }else{
                echo "<tr><td colspan=7 align=center>Data empty</td></tr>";
            }
        ?>
    </table>
</body>

```

Phù cuối cùng thì cũng xong, đây là lần thứ 2 viết lại, hiz.. nãy viết vừa xong nhấp xem lại bài viết thì bị đá giăng ra ngoài, làm mất sạch :((lại phải ì ạch viết lại. Kết quả như hình sau:

Select a file:

STT	Tên bài hát	Ca sĩ	Nhạc sĩ	Thể loại	Album
1	A time for us	Ngân Khánh - Ngô Kiến Huy	LV Phạm Duy	Nhạc Hot	Album 1 2
2	Âm nhạc của tôi	Takej Minh Huy	Young Uno	Nhạc Hot	Album Âm tôi
3	Âm nhạc của tôi	Takej Minh Huy	Young Uno	Nhạc Hot	Album My
4	Anh chỉ là hình bóng của người khác	Sỹ Luân	Sỹ Luân	Nhạc Hot	Album 1 2
5	Anh đi tìm em	Takej Minh Huy	Phúc Bỏ	Nhạc Hot	Album Âm tôi
6	Anh không hiểu	Quang Hà	Hồ Anh Dũng	Nhạc Hot	Album Mis
7	Anh mê vợ bé	Son ca - Huy Vũ	Vũ Quốc Bình	Nhạc Hot	Album Mis
8	Anh muốn thầm nói với em rằng anh yêu em	Quang Vinh	n/a	Nhạc Hot	Album Mis
9	Anh nhớ ai	Son ca	Nguyễn Văn Muối Tiểu	Nhạc Hot	Album Mis
10	Ảnh sao băng	Sỹ Luân	Sỹ Luân	Nhạc Hot	Album 1 2
11	Anh tin mình đã cho nhau một kỷ niệm	Thu Thủy - Lương Bằng Quang	Lương Bằng Quang	Nhạc Hot	Album 1 2
12	Ảnh trắng khuya	Quang Hà	Lương Bằng Quang	Nhạc Hot	Album Mis
13	Anh về	Quách An An		Nhạc Hot	Album Q.A
14	Ba ngọn nến lung linh	Hoàng Ngân-Xuân Nghi	Ngọc Lễ	Nhạc Thiếu Nhi	Album Q.A
15	Baby love	Anh Tài	Hồ Minh Phúc	Nhạc Thiếu Nhi	Album Anh Collection
16	1 2 3 chia đôi lối về	Quang Hà	Bảo Chính	Nhạc Trẻ	Album Mis
17	1 2 3 ngôi sao	Sỹ Luân	Sỹ Luân	Nhạc Trẻ	Album 1 2
18	1000 lý do anh đặt ra	Son ca	Vũ Quốc Bình	Nhạc Trẻ	Album 1 2
19	Anh xin lỗi	Lâm Chấn Hải	Cao Đình Trung	Nhạc Trẻ	Album Q.A

Đây chỉ là bài demo ý chính là nhập dữ liệu bằng file excel, nên có nhiều cái còn lỗi
thời lუმ thuộm, mong các bạn bỏ qua nhé ^_^, bạn nào chưa biết thì có thể tham

khảo , bạn nào biết rồi mà biết cách viết hay hơn thì nhớ share cho mọi người cùng học hỏi nhé.

Ai không hiểu thì có thể hỏi mình hoặc pm wa nick yà hú: thugianmotti :D

Mô hình VMC auto load Controller và model

---VMC là mô hình quen thuộc với các bạn đã học framework hay đã tìm hiểu. Đó là điều không thể thiếu để chuẩn bị bắt đầu vào học 1 framework nào đó. Hôm nay mình xin chia sẻ mô hình mà mình học được cho những bạn chuẩn bị hay đang có ý định tìm hiểu 1 framework nào đó. Trong forum có vài bài viết về mô hình này rất hay các bạn có thể tìm đọc để hiểu thế nào là mô hình MVC (Model - View - Controller) và cách hoạt động chức năng của từng file. <http://www.qhonline.info/forum/showt...co-ban-voi-php> hoặc <http://www.qhonline.info/forum/showt...c-cua-thay-huy>

---**Ưu điểm** : là tự động load controller và model gửi dữ liệu và hiển thị trong view gần giống như trong các framework

---file **Bootstrap.php** :

PHP Code:

```
<?php
class Bootstrap{
    public function __construct(){
        if(isset($_GET['url'])){
            $url=rtrim($_GET['url'], "/");
            $url=explode('/', $url);
            $c=$url[0];
        }else{
            $c="index";
        }
        $file_controller=__CONTROLLER_PATH.$c."_Controller.php";
        if(file_exists($file_controller)){
            require_once($file_controller);
        }else{
            require_once(__CONTROLLER_PATH."error_Controller.php");
            $controller=new error();
            $controller->index();
            return false;
        }
        $name_controller=$c."_Controller";
        $controller=new $name_controller;
        $controller->LoadModel($c); //autoload model
        if(isset($_url[2])){
            $controller->{$_url[1]}($_url[2]);
        }
    }
}
```

```

        }else{
            if(isset($url[1])){
                $controller->{$url[1]}();
            }else{
                $controller->index();
            }
        }
    }
}
}
?>

```

+file này có tác dụng tự động load controller và model khi người dùng tương tác tới trang
+Lưu ý:Đặt tên file controller và model có phần đầu giống nhau,và tên file giống tên class,ví dụ user_Controller.php thì class tương ứng sẽ là class user_Controller và model sẽ là user_Model.php class user_Model.Như vậy trong view sẽ có thư mục tương ứng là thư mục user,

---Trang index sẽ khởi tạo Bootstrap như sau :

PHP Code:

```

<?php
//define path
require_once("config/define_path.php");
require_once("config/define_database.php");
//end define path
require_once(__LIB_PATH."Session.php");
require_once(__LIB_PATH."Controller.php");
require_once(__LIB_PATH."View.php");
require_once(__LIB_PATH."Model.php");
require_once(__LIB_PATH."Bootstrap.php");
$bootstrap= new Bootstrap;
?>

```

---Khi người dùng truy cập vào trang <http://localhost/vmc/user/login> Bootstrap sẽ thực hiện action login trong controller user_Controller.php và load model user_Model.php để sử dụng

---Sau đây là ví dụ login với mô hình này :

+ **Chuẩn bị** :tạo bảng user (user_name,pass)

+ **file controllers/user_Controller.php** :

PHP Code:

```

<?php
class user_Controller extends Controller{
    public function __construct(){
        parent::__construct();
    }
    public function index(){
        $this->view->render("user/index");
    }
    public function login(){
        if(isset($_POST['login'])){
            if($_POST['user']=="" or $_POST['password']==""){
                $this->view-
>msg="Hay dien user hay pass";//gửi thông báo tới trang view
                $this->view->render("user/login");
            }else{
                if($this->model->login()==true){
                    Session::set("login",true);
                    Session::set("user",$_POST['user']);

```

```

        $this->view->redirect();
    }else{
        $this->view->redirect("user/login");
    }
}
}else{
    $this->view->render("user/login");
}

}
public function logout(){
    Session::set("login",false);
    Session::unset_session("user");
    $this->view->redirect();
}
}
?>

```

---File models/user_Model.php :

PHP Code:

```

<?php
class user_Model extends Model{
    public $table="user";
    public function __construct(){
        parent::__construct();
    }
    public function login(){
        $user=$_POST['user'];
        $pass=md5($_POST['password']);
        $where=array('user_name'=>$user,'pass'=>$pass);
        $this->select($this->table,$where);
        if($this->num_rows()==0){
            return false;
        }else{
            return true;
        }
    }
}
?>

```

---file Views/user/login.php : sẽ hiển thị lỗi lấy từ controller nếu có và form đăng nhập,:

PHP Code:

```

<h2>Trang login.....</h2>
<hr />
<div class="error" style="background:#FC6;color:red">
<?php
if(isset($this->msg)){
    echo $this->msg;
}
?>
</div>
<form action="" method="post">
<label>User </label><input name="user" type="text" /><br />
<label>Password </label><input name="password" type="text" /><br />
<label></label><input type="submit" value="login" name="login"/><br />
</form>

```

---Các bạn có thể down về để hiểu rõ thêm :
<http://www.mediafire.com/?3d59b8eb70x27dx>
---PS:Em xin public bài này nhé,Có vấn đề gì sai sót mong các bạn góp ý

Thủ Thuật Tăng Tốc PHP

1. Biến

- Khai báo khởi tạo biến định kiểu nhanh hơn 376% so với không định kiểu.
- Hằng chậm hơn 146% so với biến.
- Biến cục bộ nhanh hơn 9.9% biến toàn cục.

2. Hàm làm việc với chuỗi

- 'string' nhanh hơn 0.26% so với "string"
- "String" nhanh hơn 4% so với cú pháp HEREDOC
- "String\n" nhanh hơn 108% so với 'String'. "\n"
- 'String'.\$var nhanh hơn 28% so với "String\$var"
- 'string'.'\$var.' string' nhanh hơn 55% so với sprintf('string %s string', \$var)
- "\n" nhanh hơn 70% so với chr(10)
- strnatcmp() nhanh hơn 4.95% so với strcmp()
- strcasecmp() nhanh hơn 45% so với preg_match()
- strcasecmp() nhanh hơn 6.6% so với strtoupper(\$string) == "STRING"
- strcasecmp() nhanh hơn 13% so với strnatcasecmp()
- strstr(\$string, \$string1, \$string2) nhanh hơn 10% so với str_replace()
- str_replace() nhanh hơn 161% so với strstr(\$string, \$array)
- strpos() nhanh hơn 10% so với stripos()
- strpos() nhanh hơn 9.7% so với strstr()
- isset(\$str{5}) nhanh hơn 176% so với strlen(\$str) > 5
- str_replace(\$str, \$str, \$str) hai lần nhanh hơn 17% so với str_replace(array, array, string)
- list() = explode() nhanh hơn 13% so với substr(\$str, strpos(\$str))

3. Hàm làm việc với số

- ++\$int nhanh hơn 10% so với \$int++
- (float) nhanh hơn 48% so với settype(\$var, 'float')

4. Hàm làm việc với mảng

- list() = \$array; nhanh hơn 3.4% so với gán từng giá trị
- in_array() nhanh hơn 6% so với array_search

- `isset($array[$key])` nhanh hơn 230% so với `array_key_exists()`
- `!empty($array)` nhanh hơn 66% so với `count($array)`

5. Hàm xuất

- `echo` nhanh hơn 5% so với `print()`
- `echo ''` nhanh hơn 0.44% so với `echo ''`

6. Hàm, phương thức và cách thức gọi

- `call_user_func()` chậm hơn 54% so với gọi trực tiếp hàm
- `call_user_func()` chậm hơn 59% so với gọi phương thức tĩnh (static method)
- `call_user_func()` chậm hơn 65% so với gọi phương thức (của một đối tượng)
- `function()` nhanh hơn 119% so với `static::method()`
- `$this->method()` nhanh hơn 116% so với `static::method()`
- `declared static::method()` nhanh hơn 93% so với `static::method()`

7. Hàm, phương thức

- Truyền tham trị nhanh hơn 3% so với truyền tham chiếu
- Không tham chiếu (no reference) nhanh hơn 1.7% so với tham chiếu (Return by reference)

8. Làm việc với File

- `Scandir()` nhanh hơn 4% so với `opendir()`, `readdir()`, `closedir()`
- `file_get_contents()` nhanh hơn 52% so với `fopen()`, `fread()`, `fclose()`
- `file_get_contents()` nhanh hơn 39% so với `implode("\n", file())`

9. Hàm ghi bộ đệm (Cache)

- `xcache_set()` nhanh hơn 1,645% so với `file_put_contents()`
- `xcache_set()` nhanh hơn 646% so với `memcache->set()`
- `xcache_get()` nhanh hơn 1,312% so với `memcache->get()`

10. Hàm (Tổng quát)

- `if elseif else` nhanh hơn 0.78 % so với `switch`
- `@Error supression` (bỏ báo lỗi nếu có) chậm hơn 235% so với `without`
- `$_SERVER['REQUEST_TIME']` nhanh hơn 59% so với `time()`
- `min(array)` nhanh hơn 16% so với `min(int, int)`
- `require_once()` nhanh hơn 24% so với `include()`
- `require_once()` nhanh bằng `include_once()`
- `include('đường dẫn tương đối')` nhanh hơn 37% so với `include('đường dẫn tuyệt đối')`

11. Chuỗi chính quy (regular expressions)

- str_replace() nhanh hơn 40% so với preg_replace()
- ereg('regex') nhanh hơn 17% so với preg_match('/regex/')
- preg_match('/regex/i') nhanh hơn 68% so với eregi('regex')

12. Top 10 các hàm có ích nhất - đắt giá nhất

```
file_put_contents()
opendir() readdir() closedir()
scandir()
fopen() fread() fclose()
memcache->get()
memcache->set()
implode('\n', file())
include()
include_once()
file_get_contents()
```

13. Top 10 các hàm tiết kiệm thời gian nhiều nhất (biggest time savers (per call))

```
xcache_set() vs. file_put_contents()
xcache_get() vs. memcache->get()
xcache_set() vs. memcache->set()
file_get_contents() vs. fopen(), fread(), fclose()
file_get_contents() vs. implode('\n', file())
include(relative) vs. include(full_path)
require_once() vs. include()
scandir() vs. opendir() readdir() closedir()
static method vs. call_user_func()
preg_match(/regex/i) vs. eregi(regex)
```

Nội dung tiếng anh các bạn có thể tìm thấy từ:
joomlaperformance.com/articles/performance/

Code phân trang theo kiểu Google

Chào mọi người. Hôm nay mình mới tìm hiểu code dc cái phân trang giống google

Mô tả:

Giả sử có 20 trang .

1/ Nếu đang ở trang 1: 1 2 3 4 5 6 7 8 9 Next

2/ Nếu click vào trang 8 : Prev 4 5 6 7 8 9 10 11 12 Next

3/ Nếu đang ở trang 8 mà click vào trang 7 : Prev 3 4 5 6 7 8 9 10 11 Next

4/ Nếu click vào trang 20 : Prev 16 17 18 19 20

Tức là click vào trang nào thì trang đó sẽ là trang chính giữa(trường hợp này là xếp thứ 5) và page list luôn có 9 số hiển thị . Tất nhiên là code này chỉ dùng khi nào có hơn 9 trang .

Thuật toán của phân trang như sau :

a/ Dùng SQL : \$sql = "select * from table_name LIMIT \$start,\$limit"; (\$limit là số record trả về, \$start là vị trí bắt đầu dc select).

b/ Mình tìm tổng số record trong database(\$count) rồi đem chia cho \$limit để tìm ra số trang sẽ hiển thị .

c/ Mỗi khi load trang thì \$start sẽ tăng lên 1 lượng \$limit . Ví dụ, trang 1 thì lấy từ \$start =0 , trả về \$limit dòng, trang 2 select từ vị trí \$start+\$limit, trả về cũng \$limit dòng . Số record trả về không đổi , tùy theo mình muốn 1 trang hiển thị bao nhiêu record từ database . Mỗi trang thì số \$start thay đổi theo kỉ luật.

Mình tạo class pager gồm 3 function sau :

1/ Function findStart() :

Hàm này trả về \$start ở trên mình nói .

2/ Function findPages(\$count,\$limit)

Trả về số trang cần hiển thị

3/ Function pageList(\$currentPage, \$pages, \$count)

Trả về layout pageList ra . Như cái " Prev 1 2 3.... Next " ở trên đây

Code : phantrang_google.php

PHP Code:

```
<?php
class phantrang_google_pager
{
    function findStart()
    {
        if(isset($_GET['page'])){ // Lúc đầu tiên mở ra thì hok có $_GET['page']
            return $start = $_GET['page'];
        }
        else{
            return $start = 0;
        }
    }

    function findPages($count,$limit)
    {
        return $count % $limit ==0 ? $count/$limit :ceil($count / $limit);
        /* Câu trên có nghĩa : $count có chia hết cho $limit không? Nếu có thì trả về $count/$limit, nếu không thì trả về $count/$limit +1 . (ceil là hàm làm tròn trên , 2.3=3 , 5.6 =6)
        */
    }

    function pageList($currentPage,$pages,$count)//Biến $count ở đây gì ai thích sau
    {
        $space =4;
        if(!isset($_GET['middle']) || $currentPage<=5){
```

```

        //Khi mở ra thì cũng chưa có $_GET['middle']

        $middle =5;
    }
    else{
        $middle = $_GET['middle'];
    }
    $start_temp = $middle -$space;
    // Biến này chứa số đầu tiên của 9 số trên
    $end_temp =$middle+ $space;
    // Biến này chứa số cuối cùng của 9 số trên
    $i = $start_temp;
    // Chuẩn bị dùng vòng lặp để xuất ra
    if($currentPage != 1){
    // Nếu trang hiện tại không phải 1 thì xuất Prev
        $prev = $currentPage-1;
        $prev_middle = $middle-1;
        echo "<span id ='list'><a href = '$_SERVER[PHP_SELF]?middle=$prev_middle&count=$count&page=$prev'> Prev </a></span>";
    }
    //$_GET['middle'] chỉ số nằm ở giữa. $_GET['count'] chỉ tổng record trong database.
    //$_GET['page'] chỉ số trang người dùng đang đứng
    // $_SERVER['PHP_SELF'] -
    > lấy thanh địa chỉ, nó chỉ lấy tới hết chữ php thôi
        while($i) //Dùng vòng lặp hiển thị trang
    {

        if($i > $pages || $i > $end_temp){
            break;
        }
        // Nếu số của trang lớn hơn trang cuối, hay lớn hơn vị trí cuối của 9 số thì dừng cho //trang này

        if($i != $currentPage){
        //Số trang người dùng đang đứng phải hiển thị khác để người dùng biết đang đứng đâu
            if($i <= 5) {
                //5 ở đây do mình mặc định là 9 , thì vị trí chính giữa đầu tiên là 5
                echo "<span id ='list'><a href = '$_SERVER[PHP_SELF]?middle=5&count=$count&page=$i'> $i </a></span>";
                //Với ít hơn 5 đầu tiên thì middle vẫn là 5
            }
            if($i> $middle || ($i < $middle && $i > 5) ){
                echo "<span id='list'><a href = '$_SERVER[PHP_SELF]?middle=$i&count=$count&page=$i'> $i </a></span>";
            }
            //Nếu bỏ $i < $middle && $i >5 <<<<<<< bước 3 sẽ không làm được
        }
        else {

            echo "<span id ='list' >$i</span>";

        }

        $i++;
    }

```

```

}
    if($currentPage != $pages){
// Nếu không ở trang cuối thì hiển thị nút Next
        $next = $currentPage+1;
        $next_middle = $middle+1;
        echo "<span id ='list'><a href ='$_SERVER[PHP_SELF]?middle=
$next_middle&count=$count&page=$next'> Next </a></span>";
    }
}
}

```

Code: demo_google.php

PHP Code:

```

<style>
    #list{
        border : 1px solid green;
        padding:5px;
        margin: 0px 3px;
    }
    #selected{
        background: green;
    }
    a{
        text-decoration: none;
    }
    a:hover{
        color: white;
    }
    #list:hover{
        background : green;
    }
    #list:hover a{
        color:white;
    }
</style>
<?php
    require_once('phantrang_google.php');
    $conn = new mysqli('localhost','root','','news_vmc');
//Mình dùng database của bạn news_vmc .
//Link down :
    if(!isset($_GET['count'])){
// Lúc mở ra cũng chưa có $_GET['count']
        $query = 'select * from tin';
        $result = $conn->query($query);
        $count = $result->num_rows;//Trả về số record trong table
//Dùng mysqli
    }
    else{
        $count = $_GET['count'];
    }
//Đoạn code trên để tránh việc làm lại 2 câu truy vấn mỗi khi chuyển tr
ang, nó là công //dụng của biến $count trong function pageList ở bên ki
a

```

```

$page = new phantrang_google_pager;
$limit = 3;
//Mình giới hạn 3 records 1 trang
$start=$page->findStart();
//Tìm start trước
if(!isset($_GET['page'])){
    $curpage=1;
}
else{
    $curpage = $_GET['page'];
}

$stt = ($curpage-1)*3+1;
$pages = $page->findPages($count,$limit);

$sql = "select * from tin LIMIT $start,3";
$result = $conn->query($sql);

echo " <table border='1px'>";
while($rows = $result->fetch_assoc()){
    echo "<tr>";
    echo "<td>$stt</td>";
    echo "<td>$rows[id_tin]</td>";
        echo "<td>$rows[tieu_de]</td>";
        echo "<td>$rows[tom_tat]</td>";
    echo "<tr>";
    $stt++;
}
echo "</table>";
echo "<br/>";
echo $page->pageList($curpage,$pages,$count);

```

?>

Lưu ý : mình muốn thể hiện 9 trang, các bạn có thể muốn 10 11 12 trang thì sửa trong code :).

Link down database news_vmc: *

mediafirechamcom ?f9inhvd1kddn4j2 *

Hình ảnh khi code xong :)

***photobucketchamcom/albums/i366/danghaiphung169/11212.png**
photobucketchamcom/albums/i366/danghaiphung169/112122.png
photobucketchamcom/albums/i366/danghaiphung169/1121224.png*

Bye mọi người

Phân trang cơ bản, có phân đoạn không OOP

PHP Code:

```
$conn=mysql_connect("localhost","root","*****");
mysql_select_db("vd_1",$conn);
if(isset($_GET['page'])){
    $current_page=$_GET['page'];
}else{
    $current_page=1;
}
//so record tren trang
$record_in_page = 3;
//Tong record
if(!isset($total)){
    $query=mysql_query("select * from user");
    $total=mysql_num_rows($query);
}else{
    $total=$_GET['total'];
}
echo $total;
$y = $record_in_page*($current_page-1);
//Tong so trang
$total_page=ceil($total/$record_in_page);
//So trang hien thi
$roll = 3;

$query_out = mysql_query("select * from user LIMIT $y,$record_in_page")
;
while($data=mysql_fetch_assoc($query_out)){
    echo "<table>";
    echo "<tr><td>$data[username]</td><td>$data[email]</td></tr>";
    echo "</table>";
}
echo "So trang: ";
if($current_page!=1){
    echo "<a href='paging.php?total=$total&page=".(($current_page-1)."'>PRE</a>";
}

//Lay roll begin va end
if(($current_page-ceil($roll/2))<1){
    $begin=1;
    $end = $roll;
}else{
    $begin=$current_page-ceil($roll/2);
    if($current_page>=$total_page){
        $end=$total_page;
        $begin=$total_page-$roll+1;
    }else{
        $end=$begin+$roll-1;
        if($end===$current_page){
```

```

        $begin=$begin+1;
        $end=$end+1;
    }
}
}
//Xuat so trang hien thi
for($i=$begin;$i<=$end;$i++){
    if($i==$current_page){
        echo "[$i]";
    }else{
        echo "<a href='paging.php?total=$total&page=$i'> ".$i." </a>";
    }
}
if($page<$total_page){
    echo "<a href='paging.php?total=$total&page=" . ($current_page+1) . "'>
NEXT</a>";
}
mysql_close($conn);

```

Mọi người tham khảo để viết lên class nhé!

Hướng dẫn sử dụng jquery ajax lồng nhau trong ứng dụng

Thời gian qua, tôi nhận được rất nhiều câu hỏi về việc sử dụng jquery ajax lồng nhau bị lỗi. Vì thế trong bài này, tôi sẽ hướng dẫn các bạn khắc phục lỗi đó.

Ví dụ của tôi đơn giản như sau:

Có 3 liên kết. Khi click vào 1 liên kết thì sẽ xuất ra 1 nút nhấn. Nhấn tiếp vào nút đó thì sẽ ra 1 câu thông báo nào đó.

Phân tích:

Như vậy, để thực hiện thao tác này ta phải làm 2 hành động. 1 là bắt sự kiện khi click vào link. Và 2 là khi click xong sẽ xuất hiện nút, và nhấn tiếp nút thì sẽ thông báo một thông tin gì đó.

Để thực hiện ta làm như sau:

Tạo file test.html với nội dung sau:

HTML Code:

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

```

```

<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Untitled Document</title>
<script language="javascript" src="jquery.js"></script>
<script>
$(document).ready(function() {
    $("a.link").click(function() {
        id=this.getAttribute("id");
        $.ajax({
            type:"get",
            url:"check.php",
            data:"id="+id,
            async:false,
            success:function(kq) {
                $("#ketqua").html(kq);
            }
        })
    })
    $(".button").click(function() {
        id=this.getAttribute("id");
        alert("QHOnline.Info - Your link : "+id);
    })
})
</script>
</head>

<body>
    <a href="#" class="link" id="1">Link 1</a>
    <a href="#" class="link" id="2">Link 2</a>
    <a href="#" class="link" id="3">Link 3</a>
    <div id="ketqua"></div>
</body>
</html>

```

Tạo file check.php với nội dung như sau:

PHP Code:

```

<?php
$id=$_GET['id'];
echo "<button class='click' id='$id'>Click Me Link $id</button>";
?>

```

Giải thích:

Để gọi jquery ajax lồng nhau, thì chúng ta phải cho dữ liệu load trên môi trường đồng bộ. Vì sự kiện button chỉ xuất hiện khi chúng ta click vào liên kết. Do vậy, nếu các bạn để async:true. Thì dữ liệu load bất đồng bộ. Và dĩ nhiên sự kiện ở dưới sẽ gọi đồng thời mặc cho ajax muốn làm gì thì làm.

Kế tới, sự kiện click vào nút button nó có được khi nào ?. Nó có được khi người dùng click vào link. Vì thế. Sự kiện nút button phải nằm trong sự kiện click vào link đầu tiên. Nếu đặt ở bên ngoài sự kiện click link thì nó sẽ không tìm thấy được :button. Vì mới mở ra làm gì có :button cho jquery quét dữ liệu.

Vậy tổng kết lại:

- Để sử dụng jquery ajax lồng nhau ta cần chú ý:**
- + Dữ liệu load phải đồng bộ. Nghĩa là các sự kiện phải đợi nhau.
 - + Các sự kiện phải lồng trong nhau, chứ không đứng độc lập ở bên ngoài.

Hy vọng, sẽ giúp các bạn giải quyết vấn đề mà các bạn đang gặp phải.

Good Luck.

Code phân trang bằng Ajax đơn giản

Kính chào Thầy và các bạn . Ngày hôm nay em xin viết 1 Tut đơn giản về việc phân trang bằng Ajax .

Như chúng ta đã biết cái ưu điểm của Ajax là xử lý bất đồng bộ , nó không làm trang web của chúng ta phải load lại toàn bộ mà chỉ load lại ở 1 bộ phận nào đó mà chúng ta xử lý . Hôm nay , mình xin ứng dụng lại những gì đã học từ Thầy và viết code đơn giản . Có gì sai sót các bạn góp ý dùm mình .

Tut của mình chỉ đơn giản gồm có 1 combo box để chúng ta chọn thể loại . Sau khi chọn thể loại thì sẽ đổ dữ liệu trong thể loại đó ra kèm theo phân trang . Ở đây mình chưa phân nhóm , các bạn có thể góp ý xây dựng hoàn chỉnh để phân trang và phân nhóm cho hoàn chỉnh nha . Sau đây là các bước thực hiện :

Đầu tiên chúng ta tạo 1 trang có tên là theloai.php . Trong trang này là 1 form để chúng ta chọn thể loại từ cơ sở dữ liệu . Sau đó tạo 1 trang ajax.js và 1 trang process.php để xử lý như trong code bên dưới .

Các bạn tham khảo code đầy đủ ở dưới nha .

Cấu trúc thư mục như sau :

folder : includes/ : trong folder này chứa file config.php chứa các thông số cấu hình

Code :

PHP Code:

```
<?php
    $host = "localhost";
    $user_host = "root";
    $pass_host = "root";
    $db_name = "tintuc";

    $link = mysql_connect($host,$user_host,$pass_host) or die("can not
connect host");
    mysql_select_db($db_name,$link) or die("can not connect database");

?>
```

folder data/ : chứa image của tin tức

file : theloai.php

Code :

PHP Code:

```
<script language="javascript" src="ajax.js"></script>
<select size="1" name="tl" class="title" onchange="if(this.value!=0){ja
vascript:loadXMLDoc(this.value,1);}">
    <option value="0">--Chọn thể loại--</option>
    <?php
        require_once("includes/config.php");
        $sql = "select * from theloai";
        $result = mysql_query($sql);
        while($data = mysql_fetch_array($result)){
            echo "<option value='$data[idTL]'>$data[TenTL]</option>";
        }
    ?>
</select>
<div id="ketqua"></div>
```

file ajax.js :

PHP Code:

```
// JavaScript Document
function obj(){
    td = navigator.appName;
    if(td == "Microsoft Internet Explorer"){
        dd = new ActiveXObject("Microsoft.XMLHTTP");
    }else{
        dd = new XMLHttpRequest();
    }
    return dd;
}

http = obj();

function loadXMLDoc(id,p){
    document.getElementById("ketqua").innerHTML = "Đang xử lý...";
    url = "process.php?id="+id+"&page="+p;
    http.open("get",url,true);
    http.onreadystatechange=process;
    http.send(null);
}

function process(){
    if(http.readyState == 4 && http.status == 200){
        document.getElementById("ketqua").innerHTML = http.responseText;
    }
}
```

file proccess.php

code :

PHP Code:

```
<?php
    $id = $_GET['id'];
    $page = $_GET['page'];
    require_once("includes/config.php");
    $sql = "select * from tin where idTL=$id";
    $result = mysql_query($sql);
    $tongsorecord = mysql_num_rows($result);
```

```

$y = 1;
$start = ($page-1)*$y;
$tongsotrang = ceil($tongsorecord/$y);
if($tongsorecord == 0){
    echo "Không có tin nào";
}else{
?>
<script language="javascript" src="ajax.js"></script>
    <table align="center">
        <tr>
            <td class="title">STT</td>
            <td class="title">Tiêu đề</td>
            <td class="title">Hình</td>
        </tr>
    </tr>
    <tr>
        <td colspan="3">
            <div id="pt">
                <?php
                    $stt = 0;
                    $sql1 = "select * from tin where idTL=$id limit $start,$y";
                    $result1 = mysql_query($sql1);
                    while($data1 = mysql_fetch_array($result1)){
                        $stt++;
                        echo "<tr>";
                        echo "<td>$stt</td>";
                        echo "<td>$data1[TieuDe]</td>";
                        echo "<td><img src='data/$data1[UrlHinh]' title='$data1[Tom
Tat]' /></td>";
                        echo "</tr>";
                    }
                </tr>
            </div>
        </td>
    </tr>
</table>

```

PS : Đã update bài viết :)

Sử dụng JqGrid PHP

---Bài hướng này mình sẽ hướng dẫn các bạn sử dụng thư viện JqGrid PHP cơ bản nhất nhưng cũng đủ để tạo ra 1 trang quản lý dữ liệu không quá phức tạp

I/Ưu Điểm :

- Dễ dàng cài đặt và sử dụng
- Giao diện quản lý chuyên nghiệp
- Các chức năng trong bài hướng dẫn này :
 - +Thêm ,sửa , xóa dữ liệu
 - +Phân trang dữ liệu
 - +Date-picker
 - +Sorting the Data
 - +Auto-filter Search
 - +Export to Excel *
 - +Export to PDF (with Custom PDF formats) *
 - +Hỗ trợ UTF8

.....

II/ Cài đặt :

---Link demo : <http://www.trirand.net/demophp.aspx> các bạn vào đó xem demo sẽ hiểu cách sử dụng và tìm hiểu sâu hơn những vấn đề khác. Trong bài này mình chỉ tổng hợp lại những phần hay dùng nhất cho 1 trang quản lý dữ liệu. Bắt đầu nhé :

Bước 1 : Các bạn download jqsuite phiên bản mới nhất tại

<http://www.trirand.net/download.aspx> về test thử có các ví dụ rất hay và dễ hiểu

+Khi download về có 3 thư mục mà mình sử dụng trong bài này đó là thư mục js,php chứa thư viện jqgrid ,thư mục themes chứa file giao diện

Bước 2 : Tạo file index.php với nội dung :

PHP Code:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html>
  <head>
    <title>jqGrid PHP Demo</title>
    <link rel="stylesheet" type="text/css" media="screen" href="themes/redmond/jquery-ui-1.8.16.custom.css" />
    <link rel="stylesheet" type="text/css" media="screen" href="themes/ui.jqgrid.css" />
    <link rel="stylesheet" type="text/css" media="screen" href="themes/ui.multiselect.css" />
    </style>
    <script src="js/jquery.js" type="text/javascript"></script>
    <script src="js/i18n/grid.locale-en.js" type="text/javascript"></script>
    <script src="js/jquery.jqGrid.min.js" type="text/javascript"></script>
    <script src="js/jquery-ui-custom.min.js" type="text/javascript"></script>
  </head>
  <body>
    <div>
      <?php include ("grid.php");?>
    </div>
```

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

---Nạp các thư viện css và js của jqgrid

---Jqgrid có nhiều themes rất đẹp các bạn có thể download về đặt trong thư mục themes sử dụng và chú ý thay đổi đường dẫn tới file css cho phù hợp :

PHP Code:

```
<link rel="stylesheet" type="text/css" media="screen" href="themes/the
mesName/jquery-ui-1.8.16.custom.css" />
```

Bước 3 : Tạo file jq-config.php

PHP Code:

```
define('DB_DSN','mysql:host=localhost;dbname=news_vmc');
define('DB_USER','root'); // Your MySQL username
define('DB_PASSWORD',''); // ...and password
```

Bước 4 : Tập file grid.php với nội dung :

PHP Code:

```
<?php
require_once 'jq-config.php';
// include the jqGrid Class
require_once "php/jqGrid.php";
// include the driver class
require_once "php/jqGridPdo.php";
// Connection to the server
// include the autocomplete class
require_once "php/jqAutocomplete.php";
// include the datepicker class
require_once "php/jqCalendar.php";

$conn = new PDO(DB_DSN,DB_USER,DB_PASSWORD);
// Tell the db that we use utf-8
$conn->query("SET NAMES utf8");

// Create the jqGrid instance
$grid = new jqGridRender($conn);
// Write the SQL Query
$sql = 'SELECT * FROM user';
$grid->SelectCommand = $sql;
// Set the table to where you add the data
$grid->table = 'user';
$grid->setPrimaryKeyId('id');
$grid->serialKey = false;
// Set output format to json
$grid->dataType = 'json';

// Set the url from where we obtain the data
$grid->setColModel();
$grid->setUrl('gridUser.php');

// Set some grid options
$grid->setGridOptions(array(
    "caption"=>"User header",
    "rownumbers"=>true,//thứ tự các dòng 1 2 3 4 5...
    "altRows"=>true,
    "hoverrows"=>true,//hover rows có màu
    "rowNum"=>5,//số dòng mặc định hiển thị trên grid
```

```

        "rowList"=>array(5,10,15,20), //số dòng hiển thị trên grid
        "sortname"=>"user",
        "width"=>800,
        "height"=>200,
        "multiselect"=>true, //chọn nhiều rows 1 lúc
    ));

    // option row
    $grid->setColProperty('id', array(
        "editrules"=>array("required"=>true, "editable"=>"false"
    ));

    $grid->setColProperty('user', array(
        "width"=>"100",
        "editrules"=>array("required"=>true)));

    $grid->setColProperty('pass', array(
        "width"=>"100",
        "editrules"=>array("required"=>true)));

    $grid->setColProperty('email', array(
        "editrules"=>array("required"=>true)));
    $grid->setColProperty('ngay_tham_gia',
        array("formatter"=>"date",
            "formatoptions"=>array("srcformat"=>"Y-m-d", "newformat"=>"d-m-Y"),
            "editoptions"=>array("dataInit"=>
                "js:function(elm) {setTimeout(function() {
                    jQuery(elm).datepicker({dateFormat:'yy-mm-dd'});
                    jQuery('.ui-datepicker').css({'font-size':'75%'});
                },200);}");
            ));
    // Enable navigator
    $grid->navigator = true;

    // option navigator
    $grid->setNavOptions('navigator', array(
        "pdf"=>true,
        "excel"=>true,
        "add"=>true,
        "edit"=>true,
        "del"=>true,
        "view"=>false,
        "search"=>true));

    $grid->setNavOptions('edit', array(
        "closeAfterEdit"=>true,
        "editCaption"=>"Update User",
        "bSubmit"=>"Update"));

    $grid->setNavOptions('add', array("bSubmit"=>"Add"));

    // Set autocomplete only on serch form.Gợi ý khi tìm kiếm user
    $grid-
    >setAutocomplete("user", false, "SELECT DISTINCT user FROM user WHERE use
    r LIKE ? ORDER BY user", null, false ,true);

```

```

//user datetime picker
//$grid-
>setDatepicker('OrderDate', null, true, true); nếu auto hiện datetime pic
ker
$grid-
>setDatepicker('ngay_tham_gia', array("buttonIcon"=>true), true, false)
;
$grid->datearray = array('ngay_tham_gia');

// prevent some executions when not excel export
$oper = jqGridUtils::GetParam("oper");
if($oper == "pdf") {
    $grid->setPdfOptions(array(
        // enable header information
        "header"=>true,
        // set bigger top margin
        "margin_top"=>27,
        // set logo image
        "header_logo"=>"logo.gif",
        // set logo image width
        "header_logo_width"=>30,
        //header title
        "header_title"=>"Thong ke user nobitacnt",
        // and a header string to print
        "header_string"=>"by nobitacnt"
    ));
}

// Enjoy
$grid->renderGrid('#grid','#pager',true, null, null, true,true);
$conn = null;
?>

```

PS : Còn rất nhiều lời ích khác từ thư viện này các bạn quan tâm thì nghiên cứu thêm nha.Chúc thành công

Kỹ thuật seo website

Keywords

1. Từ khóa trong thẻ tiêu đề +3

Đây là một trong những nơi quan trọng nhất để đặt từ khóa và bạn phải từ có từ khóa cần Seo trong thẻ này. Bởi vì những gì được viết bên trong thẻ sẽ hiện thị trên kết quả tìm kiếm của google. Thẻ tiêu đề cần phải ngắn (Tối đa 6-7 là hiệu quả) và từ khóa seo của bạn cần phải gần đầu dòng.

2.Từ khóa trong URL +4

Từ khóa trong url sẽ giúp rất nhiều. Ví dụ trong link <http://www.huamyvan.com/Cac->

[Loai-Van...Pham-Khac.html](#), Thì từ khóa “van phong pham” là từ mà bạn muốn xếp hạng cao khi search. Bạn nhớ rằng những từ khóa không có trong trang bạn không được đặt lên URL

3. Mật độ của từ khóa trên website +3

Bạn cần kiểm tra mật độ từ khóa. Từ 3-7% cho từ khóa seo là tốt nhất. 1-2% thì ít quan trọng hơn. Và bạn cần tránh mật độ từ khóa Seo trên 10% vì nó trông như bạn đang cố ý nhồi nhét từ khóa thay vì viết một văn bản mang tính tự nhiên

4. Từ khóa trong anchor text +3

Từ khóa trong anchor text cũng rất quan trọng đặc biệt là anchor text của backlink, bởi vì nếu bạn có từ khóa anchor text từ trang website khác, điều này được xem như nhận một phiếu bầu từ trang này không chỉ về trang web của bạn nói chung mà cả về từ khóa seo của bạn.

5. Từ khóa trong các nhóm thẻ (<H1>,<H2>) +3

Đây là vị trí mà từ khóa seo của bạn cần xuất hiện nhiều. Nhưng bạn phải chú ý trong trang của bạn phải có nội dung cho các từ khóa này

6. Từ khóa seo bắt đầu 1 đoạn +2

Từ khóa bắt đầu đầu 1 đoạn thì vẫn được đề cao về tính hiệu quả mặc dù nó không quan trọng bằng từ khóa trong anchor text, title hoặc headings. Bạn không nhất thiết phải đặt trên cùng của phần nội dung, bạn vẫn có thể đặt ở giữa nhưng chú ý là bạn cần phải đặt ở đầu đoạn

7. Từ khóa trong tags <alt> +2

Con bộ của google không thể nào đọc được hình ảnh nhưng nó có thể đọc được đoạn mô tả cho hình ảnh trong tags <alt>, bởi vậy nếu bạn có hình ảnh trên web bạn cần thêm từ khóa vào tags <alt>

8. Từ khóa trong thẻ meta (Description, Keywords) +1

Rất ít quan trọng đặc biệt là với google. Yahoo! Và Bing thì vẫn dựa vào tiêu chí này, bởi vậy nếu bạn tối ưu cho Yahoo! hoặc Bing thì hãy điền đầy đủ cho tags này. Trong mọi trường hợp điền đầy đủ đều không gây ảnh hưởng gì.

9. Từ khóa gần +1

Từ khóa gần thì vẫn được tính mặc dù nó ít quan trọng hơn từ khóa chính. Ví dụ bạn cần Seo từ khóa “Điện thoại iphone” thì tại đoạn đầu bạn đặt câu “Giới thiệu điện thoại mới từ Apple” và đoạn số 3 bạn đặt câu “dòng Iphone nổi tiếng” thì 2 từ “Điện thoại” và

“iphone” được xem là từ khóa gần. Các tính từ khóa gần này chỉ áp dụng khi bạn Seo một từ khóa được ghép từ 2 từ trở lên.

10. Cụm từ khóa: +1

Ngoài các từ khóa cần Seo của bạn, Bạn có thể tối ưu hóa cho cụm từ khóa mà nó chứa những từ khóa cần SEO. Tốt nhất là bạn sử dụng những từ khóa phổ biến được gợi ý bởi google. Ví dụ để Seo từ khóa “Điện thoại” bạn có thể sử dụng các cụm từ phổ biến trong nội dung của mình như “Điện thoại giá rẻ” hay “Điện thoại di động” hoặc “Điện thoại iphone”

11. Lỗi chính tả: +0 (Lời khuyên)

Lỗi chính tả xảy ra rất nhiều và nếu bạn biết cách hướng từ khóa của mình vào những lỗi phổ biến hoặc những từ thay thế (ví dụ: Điện thoại di động và dtdd), bạn có thể có một số lượng truy cập từ những người search dạng từ khóa này. Nhưng việc bạn đặt lỗi chính tả vào website thì không tốt. Vì thế bạn không cần đặt hoặc nếu cần thì chỉ đặt trong thẻ meta

12. Tối ưu quá nhiều từ khóa -2

Khi bạn tối ưu quá nhiều từ khóa cho một trang đặc biệt là những từ khóa không liên quan. Điều này sẽ gây giảm hiệu suất cho các từ khóa khác của bạn và thậm chí là cả từ khóa chính cũng bị ảnh hưởng theo.

13. Nhồi nhét từ khóa: -3

Bất kì hành động cố ý nào về việc nhồi nhét từ khóa (hơn 10%) bạn đều có nguy cơ bị cấm từ công cụ tìm kiếm

I . Links - internal, inbound, outbound

14 .Từ khóa(Anchor text) trong backlink. +3

Từ khóa trong backlink là một trong những yếu tố quan trọng nhất để có một thứ hạng tốt cho từ khóa.

15 .Mức độ uy tín của Backlink +3

Khi bạn đặt anchor text hãy chú trọng đến chất lượng anchor text trở về website bạn. Anchor text có chất lượng là link từ những website uy tín và thông thường những trang uy tín có điểm Google PR cao.

16. Link từ những trang cùng lĩnh vực +3

Có những link từ các trang cùng lĩnh vực hoặc tối ưu từ khóa giống bạn thì rất hữu ích.

Điều này chỉ ra rằng trong cuộc cạnh tranh từ khóa bạn được đối thủ bỏ phiếu và website của bạn trở nên phổ biến hơn

17. Link từ website có tên miền .edu và .gov

Các tên miền này rất quý vì tên miền .edu và .gov được xem uy tín hơn .com, .biz, .info v.v.... Những liên kết từ tên miền .edu và .gov thường thì rất khó có được.

18. Số lượng Backlink +3

Số lượng backlink càng nhiều thì càng tốt. Nhưng uy tín của website mà bạn đặt backlink thì quan trọng hơn số lượng backlink.

19. Đặt từ khóa (anchor text) trong liên kết nội +2

Trên website của bạn bạn cần liên kết đến các link khác cũng thuộc website của bạn điều này giúp bạn kết nối các từ khóa khác trong trang và dẫn dắt bộ của google đi được hết toàn bộ khối lượng trang của bạn .

20. Nội dung xung quanh từ khóa (Anchor text) +2

Nội dung xung quanh từ khóa (anchor text) cũng rất quan trọng bởi vì điều này thể hiện tính liên kết của link và đồng thời cho thấy đây là một liên kết tự nhiên hay bạn cố ý tạo ra.

21. Tuổi của backlink +2

Backlink có thời gian tồn tại càng lâu thì càng tốt. Có nhiều liên kết trong thời gian ngắn cho thấy rằng bạn đang mua chúng thay vì link tự nhiên.

22. Link từ directories (Dmoz, Yahoo Directory v.v...) +2

Việc này rất tốt nhưng nó phụ thuộc vào Directory nào. Được liệt kê trong Dmoz, Yahoo Directory và những directory tương tự như vậy sẽ thúc đẩy rất lớn cho thứ hạng của bạn nhưng nếu có rất nhiều link từ các directories có PR thì không có ích và thậm chí có thể bị xem là spam link nếu bạn có hàng trăm hoặc hàng ngàn link như vậy

23. Số lượng link đi vào website bạn từ 1 page +1

Việc đặt ít link từ một page đến website của bạn thì tốt hơn. Bởi vì theo cách này mỗi link của bạn trở nên quan trọng hơn

24. Thiết lập khu vực liên kết từ trong trang (Named anchors) +1

Thiết lập liên kết từ trong trang thì rất hữu ích cho việc điều phối trang nhưng cũng rất hữu dụng cho Seo bởi vì bạn có thể tạo thêm một mục cho trang, một đoạn hoặc text. Ví

dụ: trong code thiết lập khu vực liên kết riêng cho trang Đọc giải thích về Anchor

25. Liên kết đến từ những trang web được tạo ra cho mục đích seo cho trang khác hoặc những trang đáng nghi ngờ +0

Những liên kết đến từ trang này không làm ảnh hưởng đến trang của bạn nếu như bạn không đặt links của những trang này vào website mình. Google xem như những liên kết đến từ trang này ngoài tầm kiểm soát của bạn do đó bạn không bị phạt nhưng trong mọi trường hợp bạn nên tránh xa những trang web được tạo ra cho mục đích seo hoặc những trang đáng nghi

26. Quá nhiều liên kết ra ngoài.

Google không thích những trang có nhiều liên kết đi ra ngoài, bởi vậy tốt nhất bạn nên giữ số liên kết ra ngoài trên website mình dưới 100 link. Có nhiều link ra ngoài không tạo lợi ích nào trong việc xếp hạng mà thậm chí còn gây nên hậu quả ngược lại

27. Quá nhiều liên kết hoặc link spam -1

Rất có hại cho thứ hạng của bạn khi bạn có nhiều link đến từ một website bởi vì nó cho thấy bạn đang mua link hoặc ít nhất là spam link. Nếu may mắn thì cũng chỉ vài link được tính cho việc SEO của bạn thôi.

28. Đặt backlink cho những trang tạo ra cho mục đích SEO hoặc những trang đáng nghi -3

Không như việc có link đến từ những trang tạo ra cho mục đích SEO hoặc những trang đáng nghi. Việc bạn đặt lại backlink trên website mình cho những trang này sẽ khiến website của bạn bị mất từ khóa hoàn toàn. Bạn cần phải định kì kiểm tra tình trạng của những trang đặt link trên trang của bạn. Ví dụ (Bạn cố ý tạo ra 5 blog wordpress hay blogspot với nội dung không liên quan để Seo cho một từ khóa chính đến website bạn, và khi bạn đặt lại backlink cho 5 blog này trong website bạn sẽ khiến website bạn rơi vào tình trạng mất từ khóa hoàn toàn.)

29. Liên kết vòng trá hình -3

Liên kết vòng xảy ra khi trang A link đến trang B, trang B link đến trang C và trang C link trở về trang A. Đó là ví dụ đơn giản về liên kết thành vòng và ngoài ra có những trường hợp khác nữa. Liên kết vòng trong như là một loại liên kết trao đổi trá hình và bị google phạt nặng

30. Liên kết link cho chiều rộng pixel nhỏ

Khi bạn có đặt link trên website mình có kích thước 1 pixel hoặc chiều rộng như vậy thì điều này được xem như là vô hình với con người và bạn đang cố gắng qua mặt bộ máy tìm kiếm.

II. Metatags

31. <Description> metatag +1

Metatags thì đang trở nên ít quan trọng hơn đối với google nhưng nó vẫn được tính. Sử dụng metatag <Description> để viết mô tả cho website của bạn. Ngược lại đối với Yahoo và Bing nó vẫn rất quan trọng. Đối với google đôi khi bạn thấy sự xuất hiện của mô tả trong kết quả tìm kiếm

32. <Keywords> metatag +1

Metatag <Keywords> không được chú ý bởi google nhưng nó vẫn được quan tâm bởi Bing và Yahoo. Chiều dài của metatag <Keywords> khoảng 10 -20 từ là tốt nhất. Không nên nhồi nhét từ khóa mà bạn không có trong nội dung việc đó có hại cho thứ hạng của bạn.1

33. <Language> metatag +1

Nếu website của bạn có ngôn ngữ cụ thể thì đừng để tab này trống. Mặc dù công cụ search đủ thông minh để xác định ngôn ngữ trên trang của bạn thay vì dựa vào sự khai báo trên thẻ meta <language> nhưng công cụ search vẫn xem xét thẻ này.

34. <Refresh> metatag

Metatag refresh là một cách để chuyển lượng khách từ một website đến 1 website khác. Chỉ làm việc này nếu bạn cần duy chuyển trang web của mình đến một tên miền mới và bạn cần tạm thời chuyển hướng khách truy cập. Nếu sử dụng trong một thời gian dài nó sẽ làm ảnh hưởng đến bản xếp hạng của bạn. Trong mọi trường hợp sử dụng chuyển hướng 301 là tốt nhất.

III. Nội Dung

35. Nội dung độc đáo +3

Có nhiều nội dung (liên quan đến ngữ cảnh website và khác với nội dung từ website khác cả về tiêu đề lẫn từ ngữ) thì sẽ giúp tăng thứ hạng của bạn 1 cách thật sự

36. Tần số thay đổi nội dung

Thay đổi thường xuyên thì được đánh giá rất cao. Sẽ rất tốt nếu bạn liên tục thêm nội dung mới cho website

37. Kích thước của từ khóa : +2

Khi 1 từ khóa trong văn bản có font lớn những từ còn lại đều đó khiến cho nó đáng chú ý hơn do đó nó quan trọng hơn phần còn lại của đoạn text. Cách tính điểm này vẫn áp dụng

cho thẻ (<H1>,<H2>). Thông thường từ quan trọng thì sử dụng font chữ lớn hơn phần còn lại của đoạn text.

38. Định dạng từ khóa +2

Tô đậm hoặc nghiêng là một cách để nhấn mạnh sự quan trọng các từ và cụm từ quan trọng. Tuy nhiên bạn chỉ sử dụng khi muốn nhấn mạnh nếu không nó sẽ mang tác dụng ngược lại.

39 . Tuổi của nội dung +2

Những nội dung vừa mới cập nhật thì được google thích hơn (Bạn nên thường xuyên cập nhật nội dung mới)

40. Kích thước file +1

Trang dài thì không được google thích, hoặc ít nhất là bạn có thể đạt thứ hạng từ khóa tốt hơn nếu bạn sử dụng 3 trang ngắn hơn thay vì tập trung cho 1 trang dài mà chỉ đề cập một chủ đề. Vì vậy bạn nên chia trang dài của mình thành những trang nhỏ hơn.

41. Phân tách nội dung -2

Theo quan điểm marketing thì việc phân tách nội dung (dựa trên IP hoặc trình duyệt v.v...) thì rất tuyệt. Nhưng đối với Seo thì bất lợi bởi vì khi bạn có một Url không liên quan đến nội dung thì hệ thống search sẽ bị nhầm lẫn nội dung trong trang của bạn.

42. Thiết kế và viết code nghèo nàn -2

Bộ máy tìm kiếm nói rằng họ không muốn một website thiết kế và viết code nghèo nàn, mặc dù chưa có trang web nào bị cấm vì code lộn xộn hoặc hình ảnh xấu nhưng khi bạn thiết kế xấu và viết code lộn xộn thì website bạn có thể không được index hết bởi vậy việc ý thức kém về thiết kế và viết code có thể gây hại cho website bạn.

43. Nội dung bất hợp pháp -3

Sử dụng nội dung có bản quyền của người khác mà không được cho sự phép hoặc nội dung thúc đẩy hành vi phạm pháp có thể khiến bạn bị đuổi khỏi công cụ tìm kiếm

44. Ẩn văn bản với người đọc -3

Đây là cách Seo mũ đen. Khi con bộ tìm kiếm khám phá ra bạn có một đoạn text đặc biệt được ẩn với người xem thì bạn sẽ bị phạt

45. Cloaking -3

Kỹ thuật cloaking trong SEO? Cloaking là một kỹ thuật SEO mà giúp cho nội dung của

site dưới mắt các Spiders của các Search Engines (cỗ máy tìm kiếm) khác với nội dung mà khách truy cập site thấy. Điều này thường được thực hiện bằng cách sẽ điều chỉnh hiện nội dung tùy theo IP truy cập website.

46. Doorway pages -3

Tạo ra những trang khác với mục đích lừa nhện rằng webstie có liên quan đến nội dung này trong khi thực tế thì không. Việc này sẽ bị phạt rất nặng từ google

47. Nội dung bị trùng lặp -3

Khi bạn có nhiều nội dung giống nhau trên nhiều trang của website bạn việc đó sẽ bị phạt và không làm có website của bạn trở nên lớn hơn. Đồng thời hãy hạn chế sự trùng lặp nội dung với

IV. Thành phần khác

48. JavaScript +0

Nếu sử dụng như là để hỗ trợ tính năng cho web thì không có vấn đề gì. Nhưng nếu bạn hiển thị nội dung thông qua JavaScript điều này sẽ rất khó cho nhện hoạt động và nếu code JavaScript được viết rất lộn xộn thì nhện không thể nào đi theo đúng hướng website muốn và bạn sẽ làm ảnh xấu.

49. Hình ảnh trong text +0

Chỉ viết text trong web thì rất nhàm chán nhưng có nhiều hình ảnh và không có text thì được xem như là lỗi trong SEO. Bạn nhớ luôn luôn cung cấp thẻ <Alt> để mô tả cho hình ảnh nhưng không nên nhồi nhét từ khóa và cung cấp thông tin không liên quan.

50. Video +0

Nếu bạn không có tapescript chứa Video thì google sẽ không lập cập nhật video vào công cụ tìm kiếm

51. Sử dụng hình ảnh cho việc liên kết link -1

Sử dụng hình ảnh để liên kết thì rất có hại, đặc biệt là khi bạn không thêm tab <alt>. Kể cả khi bạn thêm tab <alt> thì nó cũng vẫn được xem là có hại. Bởi vậy bạn chỉ sử dụng hình ảnh để điều hướng chỉ khi điều này thực sự quan trọng đối với việc bố trí thiết kế trong trang web

52. Frames -2

Fames thì rất có hại cho SEO. Và bạn chỉ dùng khi thực sự cần thiết

53. Flash -2

Bộ tìm kiếm không đánh chỉ mục cho nội dung Flash bởi vậy bạn luôn nhớ đặt nội dung cho thẻ <alt>

54. Flash trên trang chủ -3

Cách này dường như hiện tại đã không còn dùng. Nhưng có một Flash trên trang chủ và không có có HTML thì xem như là bạn chẳng Seo gì cả.

V. Domains, URLs, Web Mastery

55. Url chứa nhiều thông tin +3

Điều này rất quan trọng đặt biệt đối với Yahoo và Bing

56. Khả năng truy cập web +3

Một vấn đề cơ bản mà bạn thường bỏ quên. Đó là nếu như website của bạn (hoặc một số trang trong website) bị hư link, hoặc lỗi 404, hoặc đặt mật khẩu bảo mật thì đơn giản là những trang đó sẽ không được đánh chỉ mục.

57. Sitemap +2

Rất tốt nếu bạn có sitemap cho website vì bộ tìm kiếm đặc biệt quan tâm loại file này. Bạn có thể làm theo định dạng HTML hoặc làm theo định dạng của google

58. Độ lớn website +2

Bộ tìm kiếm rất thích những website lớn , càng lớn càng tốt. Không có website nào bị phạt vì có trên 10 000 trang cả.

59. Tuổi của website +2

Trang web có thời gian tồn tại lâu thì được đánh giá cao hơn là một website mới được thành lập.

60. Giao diện trang +2

Giao diện trang cung quan trọng cho thứ hạng bởi vì khi trang phù hợp với 1 giao diện, điều này giúp tăng thứ hạng của tất cả các trang liên quan đến giao diện này.

61. Vị trí tập tin trong trang web +1

Vị trí tập tin trong website càng gần với thư mục gốc có xu hướng xếp hạng tốt hơn so với tập tin cách xa thư mục gốc

62. Domain và subdomain +1

Có một domain riêng biệt thì tốt hơn. Ví dụ sử dụng tên miền dienthoai.com tốt hơn dienthoai.logspot.com

63. Loại tên miền +1

Loại tên miền vẫn được quan tâm theo cách khác nhau. Thông thường người ta sử dụng tên miền .com. Loại tên miền này tốt hơn .biz, .info hay .net. Nhưng loại tên miền mạnh nhất là .edu hoặc .org

64. Dấu gạch nối trong url +1

Dấu gạch nối giữa các từ trong url giúp dễ đọc và tăng điểm SEO. Điều này áp dụng cho cả tên miền chính và phần còn lại của tên miền VD: dien-thoai.com/dien-thoai-nokia.html

65. Chiều dài của url +0

Nói chung không quan trọng nhưng nếu Url quá dài thì nó bắt đầu trông giống như spam. Chính vì vậy url không nên dài hơn 10 từ (3-4 từ cho tên miền chính và 6-7 từ cho phần còn lại của tên miền)

66. Địa chỉ IP 0

Không thành vấn đề nếu như bạn chia sẻ host hoặc dùng host miễn phí. Nhưng bạn sẽ bị vào danh sách đen nếu bạn spam hoặc thực hiện các hành động bất hợp pháp.

67. Đặt quảng cáo Adsense 0

Đặt Adsense google thì không có liên quan gì đến SEO cả. Chỉ giúp bạn tăng thu nhập chứ không giúp bạn tăng vị trí từ khóa.

68. Quảng cáo google Adwords 0

Giống như adsense, Adwords không hề ảnh hưởng gì đến SEO ngoài việc giúp bạn có lưu lượng người viếng thăm website nhiều hơn

69. Thời gian chết của host -1

Thời gian chết của host liên quan trực tiếp đến khả năng tiếp cận bởi vì nếu một site thường xuyên chết thì bộ không thể index. Trong thực tế chỉ có một yếu tố đó là nhà cung cấp host không đáng tin cậy hoặc thời gian hoạt động ít hơn 97-98%.

70. Url động -1

Bộ tìm kiếm thích url tĩnh. Mặc dù bạn thấy có rất nhiều url động vẫn nằm trên top. Nhưng như vậy vẫn không tốt đặc biệt là url động trên 200 kí tự. Trong mọi trường hợp bạn nên sử dụng công cụ để chuyển url động về url tĩnh để thân thiện với SEO. Ví dụ : chuyển url động thoitrang5s.com/category.php?id=1 về url tĩnh thoitrang5s.com/vay-dam.html

71. Session Ids -2

Sử dụng session ids thì không tốt hơn việc bạn sử dụng url động. Không được dùng session ids cho thông tin mà bạn muốn đánh chỉ mục bởi bộ tìm kiếm vì việc này dễ bị hiểu nhầm là bị trùng nội dung. Ví dụ: 3 link session ids dưới đều trỏ về một nội dung với link đầu tiên là link gốc

Dien-thoai.com/nokia-1200.html

Dien-thoai.com/nokia-1200.html?id=abcd

Dien-thoai.com/nokia-1200.html?id=abcdfe

72. Cấm trong file robots.txt -2

Nếu một phần của trang bị cấm điều này sẽ ảnh hưởng đến phần còn lại của trang bởi vậy bộ tìm kiếm sẽ ít đến trang bạn hơn. Chính vì thế nếu phần nào trong website bạn không cần index cứ khai báo từ chối trong robots.txt

73. Chuyển hướng (301 and 302) -3

Không áp dụng cách này thì việc chuyển hướng sẽ làm cho bạn bị thiệt hại nhiều. Vì một chuyển hướng có thể được xem như là một kĩ thuật mũ đen khi khách truy cập bị chuyển hướng đến 1 trang khác

Phân trang đơn giản với JQuery

---Phân trang là hình thức mà rất nhiều web hiện nay sử dụng, phân trang cũng không quá phức tạp, mình xin chia sẻ code phân trang sản phẩm của mình có sử dụng JQuery đơn giản, DEMO : <http://nobitacnt.tk/shop/test/>

---class sản phẩm :

PHP Code:

```
<?php  
require_once ("cl_connectDb.php");
```

```

class sanpham extends connectdb{
    public function __construct(){
        $this->connect();
    }
    public function list_sp($start,$limit){//hiển thị sản phẩm
        $sql="select * from sanpham order by id_SP DESC LIMIT $start , $
limit ";
        $this->query($sql);
        $data=array();
        if($this->num_rows()!=0){
            while($row=$this->fetch()){
                $data[]=$row;
            }
        }else{
            $data=0;
        }
        return $data;
    }
}
?>

```

---class phân trang :

PHP Code:

```

<?php
require_once("cl_connectDb.php");
class phantrang extends connectdb{
    public $page;
    public $star;
    public $display;
    public $id;
    public function __construct(){
        $this->connect();
        $this->get_start();
        $this->display=12;
    }
    public function get_page(){
        if(isset($_GET['page'])){
            $this->page=$_GET['page'];
        }else{
            $sql="select * from sanpham";
            $this->query($sql);
            if($this->num_rows()>$this->display){
                $this->page=ceil($this->num_rows()/$this->display);
            }else{
                $this->page=1;
            }
        }
    }
    public function get_start(){
        if(isset($_GET['start'])){
            $this->star=$_GET['start'];
        }else{
            $this->star=0;
        }
    }
    public function out_put(){

```

```

echo "<ul class='nav'>";
if($this->page>1)
{
    $next=$this->display+$this->star;
    $prev=$this->star-$this->display;
    $hientai=($this->star/$this->display)+1;
    if($hientai !=1)
    {
        echo "<li><a href='javascript:void(0)' class='pt' title='".
$this->page.'" id='". $prev.'"> Preview </a></li>";
    }

    for($i=1;$i<=$this->page;$i++)
    {
        if($hientai !=$i)
        {
            echo "<li><a href='javascript:void(0)' class='pt' title='".
.$this->page.'" id='". $this->display*($i-1)."'> $i </a></li>";
        }
        else
        {
            echo "<li class='current'>$i</li>";
        }
    }

    if($hientai !=$this->page)
    {
        echo "<li><a href='javascript:void(0)' class='pt' title='".
$this->page.'" id='". $next.'"> Next </a></li>";
    }
}
else{
    echo "<li class='current'>1</li>";
}
echo "</ul>";
}
?>

```

---file js ajax.js :

PHP Code:

```

$(document).ready(function(){
    $("a.pt").click(function(){
        var page=$(this).attr("title");
        var start=this.id;
        $.ajax({
            type : "GET",
            url : "index.php?page="+page+"&start="+start,
            datatype: "html",
            beforeSend: function(){
                $('#load').html("LOADING...");
            },
            success: function(html){
                $('#load').text('');
                $('#kq').html(html);
            }
        });
    });
}

```

```

    })
})
và cuối cùng là file index :
PHP Code:
<html>
<head>
<link rel="stylesheet" type="text/css" href="style.css" />
<script type="text/javascript" src="http://ajax.googleapis.com/ajax/lib
s/jquery/1.4.2/jquery.min.js"></script>
<script type="text/javascript" src="ajax.js"></script>
</head>
<body>
<?php
echo "<div id='load'></div>";
echo "<div id='kq'>";
require_once("cl_sanpham.php");
require_once("cl_phantrang.php");
//phan trang
$pt=new phantrang;
$limit=$pt->display;
$start=$pt->star;
$pt->get_page();
//end phân trang
$sanpham=new sanpham;
$data=$sanpham->list_sp($start,$limit);
if($data!=0){
foreach($data as $row){
    $ten_sp=$row['name_SP'];
    echo $ten_sp."<br />";
}
    echo "<div id='phantrang'>";
    $pt->out_put();
    echo "</div>";
}else{
    echo "EMPTY DATA";
}
echo "</div>";
?>
</body>
</html>

```

Performance Tips trong PHP - Part 1

How are you today ?

Are you ok ? I hope so.



Are you happy ? I hope so .



Now, we will talking about **Perfromance Tips in PHP**

Tại sao khi xây dựng 1 class chúng ta cần phải có setname, getname ?

Thật sự điều đó không cần thiết, nói chung là tùy trường hợp !

Sau đây là 1 ví dụ minh hoạ:

PHP Code:

```
class hello {
    public $name = "";
    public function setname($a){
        $this->name = $a;
    }
    public function getname(){
        return $this->name;
    }
}

$doc = new hello();
```

Nếu theo 1 chuẩn thì tối ngày các bạn cứ set xong rồi get ?

PHP Code:

```
$doc->setname("Yachi");
$b = $doc->getname();
```

Nó thật sự là vô bổ phải ko ?

Thay vì làm như vậy, chúng ta làm thế này vậy , cái này sẽ run up 100% faster , cái này

ông Google bảo đảm đấy



PHP Code:

```
$doc->name = "Emiko";
$n = $doc->name;
```



Nói suông thì các bạn sẽ ko tin, thử xem nào

PHP Code:

```
class hello {
    public $name = "";
    public function setname($a){
        $this->name = $a;
    }
    public function getname(){
        return $this->name;
    }
}
```

```

$doc = new hello();
for($i=0; $i<=10; $i++){
    $a = microtime(true);
    for($j=0; $j<100000;$j++){
        $doc->name = "Emiko";
        $n = $doc->name;
    }
    echo microtime(true) - $a;
    echo "<br />";
}
echo $n;

echo "<br /><hr /><br />";

for($i=0; $i<=10; $i++){
    $a = microtime(true);
    for($j=0; $j<100000;$j++){
        $doc->setname("Yachi");
        $b = $doc->getname();
    }
    echo microtime(true) - $a;
    echo "<br />";
}
echo $b;

```

p/s : ở đây là micro nhá, nhưng trong vòng lặp for mình chỉ dùng 5 số 0, bởi vì đôi khi > 6 số 0 thì browser sẽ báo lỗi (**Maximum execution time of 30 seconds exceeded**), và tất nhiên, bao nhiêu số 0 cũng kết quả như nhau thôi 🐷

Bây giờ, thì tin rồi chứ 🐷. Hãy thay đổi cách code của mình lại nào

Những lớp và phương thức trừu tượng trong PHP 5.

Những lớp và phương thức trừu tượng trong PHP 5.

- Một lớp trừu tượng không cho phép tạo một đối tượng thể hiện lớp trừu tượng đó (an instance of a class).
- Một lớp chứa phương thức trừu tượng phải là trừu tượng.
- Một phương thức trừu tượng thì không được định nghĩa hành vi.
- Khi kế thừa một lớp trừu tượng, tất cả phương thức trừu tượng trong lớp cha phải được định nghĩa bởi lớp con. Nếu phương thức trừu tượng được định nghĩa là protected, hàm

hành vi phải được định nghĩa protected hay public mà không được định nghĩa private.
Ví dụ đơn giản lớp trừu tượng:

PHP Code:

```
<?php
abstract class AbstractClass //Khai bao mot lop truu tuong.
{
    public $number1;
    public $number2;
    //Khai bao phuong thuc truu tuong.
    abstract protected function operationName($x,$y);
    abstract protected function getValue();
    //Khai bao phuong thuc chung
    public function printOut()
    {
        echo $this->getValue();
    }
    public function setValue($x,$y)
    {
        $this->number1=$x;
        $this->number2=$y;
    }
}
class AdditionClass extends AbstractClass //Ke thua lop truu tuong
{
    // Dinh nghia hanh vi cua phuong thuc truu tuong.
    protected function getValue()
    {
        $result=$this->number1+$this->number2;
        return $result;
    }
    public function operationName($x,$y)
    {
        return "Day la phep toan Tong: $x+$y=";
    }
}

class SubtractionClass extends AbstractClass //Ke thua lop truu tuo
ng.
{
    // Dinh nghia hanh vi cua phuong thuc truu tuong.
    protected function getValue()
    {
        $result=$this->number1-$this->number2;
        return $result;
    }
    public function operationName($x,$y)
    {
        return "Day la phep toan Hieu: $x-$y=";
    }
}

$a= new AdditionClass;
$x=7;$y=7;
echo $a->operationName($x,$y);
$a->setValue($x,$y);
$a->printOut();
```

```

echo"<br />";

$b= new SubtractionClass;
$x=10;$y=7;
echo $b->operationName($x,$y);
$b->setValue($x,$y);
$b->printOut();
?>

```

Kết quả:

Day la phép toán Tổng: 7+7=14

Day la phép toán Hiệu: 10-7=3

Đây là ví dụ mình viết trong phạm vi phi kiến thức hạn hẹp của mình mong các bạn đóng góp ý kiến.:)

Mô hình Multi Modules , Multi Templates và Multi Database

*Ngồi buồn quá không có gì làm lên viết 1 bài hướng dẫn để hiểu kỹ và khi nào quên vào đọc lại.

A/Multi Modules

I/ Ý tưởng:

---Mình cũng đang tìm hiểu Zend frameword thấy nó có sử dụng 1 mô hình multi modules rất hay và linh hoạt lên có ý tưởng sử dụng ngay mô hình này trong các ứng dụng thực tế khi các bạn sử dụng nhiều modules.Nó chạy theo link dạng '**modules/controller/action/id**' Và mình cũng đã cấu hình test thử trên mô hình này theo ý muốn

II/Uu điểm :

- Dễ làm việc theo nhóm mỗi người viết 1 modules mà không liên quan tới nhau
- Dễ dàng thêm hay xóa 1 modules trong ứng dụng mà không làm ảnh hưởng tới các modules khác
- Tự động load controller và model trong module hiện tại
- Tự động load file css+js trong modules hiện tại
- Tự động load các file block trong modules hiện tại

III/Mô hình multi modules :

-Mô hình có dạng cấu trúc như sau :

PHP Code:

```

Root /
---application/
-----Bootstrap.php
-----configs/
-----modules/

```

```

-----admin/
-----controllers/
-----modles/
-----views/
-----forms/
-----blocks/
-----shopping/
-----controllers/
-----modles/
-----views/
-----forms/
-----blocks/
-----news/
-----controllers/
-----modles/
-----views/
-----forms/
-----blocks/

---public/
-----templatest/
-----admin/
-----images/
-----css/
-----js/
-----flash/
-----shopping/
-----images/
-----css/
-----js/
-----flash/
-----news/
-----images/
-----css/
-----js/
-----flash/
-----upload/
---library/
---.htaccess/
---define.php/
---index.php/

```

---Đây là mô hình mình đang sử dụng nhìn vào cấu trúc chắc các bạn đã hiểu ý nghĩa các thư mục rồi.

---File Bootstrap.php :

PHP Code:

```

<?php
class Bootstrap{
    public static $module = NULL;//lấy tên module hiện tại
    public static $controller = NULL;//lấy tên controller hiện tại
    public static $action = NULL;//lấy action hiện tại
    public static $id = NULL;//lấy id hiện tại
    private static $instance = NULL;

    public function __construct(){
        if(self::$instance == NULL){
            self::$instance = new Bootstrap;
        }
    }
}

```

```

    }

    return self::$instance;
}

public static function run(){
    Session::__init();
    if(isset($_GET['url'])){
        $url = $_GET['url'];
        $url = rtrim($url, "/");
        $url = explode("/", $url);
        self::$module = $url[0];
    }else{
        self::$module = 'default';
    }

    if(isset($url[1])){
        self::$controller = ucfirst($url[1]); //bien doi chu cai dau
        tient hanh chu hoa
    }else{
        self::$controller = 'Index';
        self::$action = 'IndexAction';
    } //end if isset $url[1];

    //lấy file controller hiện tại
    $modules_path = MODULES_PATH . self::$module; //duong dan toi mo
    dules hien tai
    $name_controller = self::$controller . 'Controller';
    $file_control = MODULES_PATH . self::$module . '/controllers/'
    . $name_controller . '.php';
    if(file_exists($file_control) && file_exists($modules_path)){
        require_once($file_control);
    }else{//chay controller error
        $file_control = MODULES_PATH . 'default/controllers/ErrorC
        ontroller.php' ;
        $name_controller = 'ErrorController';
        require_once($file_control);
    }
    $c = new $name_controller;

    $c->LoadModel(self::$controller); //auto load model

    if(isset($url[2])){
        if(isset($url[3])){
            self::$id=$url[3];
            self::$action = ucfirst($url[2]).'Action';
            $c->{self::$action}(self::$id);
        }else{
            self::$action = $url[2].'Action';
            $c->{self::$action}();
        }
    }else{
        $c->IndexAction();
    }
}

```

```
}  
}  
?>
```

Tìm hiểu thư viện Php_filter

A.PHP_FILTER là gì ?

Hiện nay, hầu hết mọi ứng dụng đều có sử dụng dữ liệu từ những nguồn ngoài không an toàn bao gồm :

- Cookies
- Dữ liệu nhập vào từ form
- Web services data
- Biến server
- Các kết quả trả về từ database

Vì thế , PHP filters được sử dụng để lọc những loại dữ liệu ấy .

Để sử dụng filter chúng ta cần dùng những hàm sau đây:

filter_var(biến, kiểu lọc) : Lọc biến với kiểu lọc chỉ định

filter_var_array(biến, mảng_kiểu_lọc): Lọc nhiều biến với nhiều kiểu lọc chỉ định khác nhau

filter_input(Loại_method_trong_Form, tên_field, kiểu lọc) : Lọc từ 1 form input với kiểu lọc chỉ định

filter_input_array(Loại_method_trong_Form, tên_field, mảng_kiểu_lọc): Lọc nhiều form input với nhiều kiểu lọc chỉ định khác nhau

B.CÁCH SỬ DỤNG

Kiểu lọc:

Có 2 loại kiểu lọc là : SANITIZE và VALIDATE

a/ SANITIZE:

- Kiểu lọc này dùng để cho phép hoặc không cho phép một số kí tự trong chuỗi
- Luôn trả về chuỗi .
- Các loại FILTER_SANITIZE cần nhớ :

FILTER_SANITIZE_NUMBER_INT: Chỉ cho phép số và các phép toán + - xuất hiện trong chuỗi

FILTER_SANITIZE_STRING : Bỏ qua hoặc mã hóa những kí tự không mong muốn và các thẻ

FILTER_SANITIZE_MAGIC_QUOTES: Thêm \ vào trước những kí tự chỉ định gồm: ' " \ NULL

FILTER_SANITIZE_SPECIAL_CHARS: Thay thế các ký tự < > & và các kí tự có mã ASCII bé hơn 32

FILTER_SANITIZE_EMAIL: Bỏ qua các kí tự không hợp pháp trong 1 email: " \$ () , : ; < > \

FILTER_SANITIZE_URL : Bỏ qua các kí tự không hợp pháp trong 1 đường dẫn . Cho phép hầu hết các kí tự thông dụng

Ví dụ:

PHP Code:

```
<?php
    $test_number_int = '5+4ab-7bc';
    var_dump(filter_var($test_number_int, FILTER_SANITIZE_NUMBER_I
NT));
//Ta được chuỗi '5+4-7'
    $test_string = '<b>ABC</b>';
    var_dump(filter_var($test_string, FILTER_SANITIZE_STRING));
//Ta được chuỗi 'ABC'

    $test_magic_quotes = "Yours ' s car ";
    var_dump(filter_var($test_magic_quotes, FILTER_SANITIZE_MAGIC_
QUOTES));
//Ta được chuỗi " 'Yours \' car ";

    $test_special_chars= "Bạn <Lovely> & <Beautiful> ";
    var_dump(filter_var($test_special_chars, FILTER_SANITIZE_MAGIC
_QUOTES));
//Khi click phải chọn view source trang web bạn sẽ thấy chuỗi " Bạn &lt
; Lovely &gt; & & Beautiful &gt;";

    $test_email = "online(mean)miss;ing'yo\u@yahoo.com ";
    var_dump(filter_var($test_email,FILTER_SANITIZE_EMAIL));
//Trả về chuỗi "onlinemeanmissing'you@yahoo.com"

    $test_url = "http://www.w3school&ls.co&m ";
    var_dump(filter_var($test_email,FILTER_SANITIZE_EMAIL));
```

```
//Trả về chuỗi "http://www.w3school.com"
?>
```

Lưu ý : SANITIZE chỉ có nhiệm vụ lược bỏ các thành phần không mong muốn trên chuỗi hợp lệ lẫn hok hợp lệ .

b/VALIDATE

- Kiểu lọc này xác định xem 1 chuỗi có đúng format của nó hay không
- Trả về data nếu hợp lệ, FALSE nếu thất bại
- Các loại FILTER_VALIDATE cần nhớ :

FILTER_VALIDATE_INT : xác nhận tính hợp lệ của 1 con số nguyên. Bên cạnh đó có các options như : min_range(giá trị nhỏ nhất cho phép) , max_range (giá trị lớn nhất cho phép)

FILTER_VALIDATE_FLOAT : xác nhận tính hợp lệ của 1 con số thực

FILTER_VALIDATE_EMAIL : xác nhận tính hợp lệ của 1 email

FILTER_VALIDATE_URL : xác nhận tính hợp lệ của 1 url

1 số FLAGS của FILTER_VALIDATE_URL:

FILTER_FLAG_SCHEME_REQUIRED:

chẳng hạn <http://example.com>

FILTER_FLAG_HOST_REQUIRED:

Yêu cầu URL có tên host chẳng hạn <http://www.example.com>

FILTER_FLAG_PATH_REQUIRED:

Yêu cầu URL có đường dẫn sau domain chẳng hạn <http://www.example.com/index/>

FILTER_FLAG_QUERY_REQUIRED:

Yêu cầu URL có chuỗi .chẳng hạn <http://www.example.php?id=5>

Ví dụ:

PHP Code:

```
<?php
    $test_int =15;
    $options = array('options'=>array('min_range' =>30, 'max_range'=>50));
    var_dump(filter_var($test_int,FILTER_VALIDATE_INT, $options))
;
//Trả về FALSE vì 15 không thuộc đoạn [30;50]

    $test_email = 'example@';
    var_dump(filter_var($test_email, FILTER_VALIDATE_EMAIL));
// Trả về FALSE .

    $test_url1 = 'http://example.com';
    $test_url2 = 'http://example.com/index/abc';
    var_dump(filter_var($test_url1,FILTER_VALIDATE_URL));
// Trả về 'http://example.com'
    var_dump(filter_var($test_url2, FILTER_VALIDATE_URL, FILTER_FLAG_PATH_REQUIRED));
// Trả về 'http://example.com/index/abc'
    var_dump(filter_var($test_url1, FILTER_VALIDATE_URL, FILTER_FLAG_PATH_REQUIRED));
```

```
// Trả về FALSE vì $test_url1 không có đường dẫn sau domain
?>
```

Chúng ta tìm hiểu về cách sử dụng các hàm còn lại.

Tạo 2 file : form.html và filter.php

form.html:

HTML Code:

```
<form action = 'filter.php' method = 'post'>
Name: <input type = 'text' name = 'name' / > <br/>
Age: <input type = 'text' name = 'age' / > <br/>
Email: <input type = 'text' name = 'email' / > <br/>
<input type = 'submit' value = 'Send' />
</form>
```

filter.php

PHP Code:

```
<?php
if(filter_has_var(INPUT_POST, 'comments')){
    echo ' Có field tên là comments trong form input ';
}
else
    echo 'Không có field tên comments trong form input<br/>';

$name = filter_input(INPUT_POST, 'name', FILTER_SANITIZE_STRING);
$options = array('options'=>array('min_range' =>18, 'max_range'
=>80));
if(!filter_input(INPUT_POST, 'age', FILTER_VALIDATE_INT, $options)
){
    echo ' Tuổi của bạn phải >18 và < 80 <br/>';
}
else{
    $age = filter_input(INPUT_POST, 'age', FILTER_VALIDATE_INT,
$options);
    if(!filter_input(INPUT_POST, 'email', FILTER_VALIDATE_EMAIL)){
        echo ' Email đã sai format<br/>';
    }
    else{
        $age = filter_input(INPUT_POST, 'email', FILTER_VALIDATE_EM
AIL);
    }
}
/*Đoạn code trên kiểm tra input của người dùng, nếu email và tuổi hợp l
ệ thì mới tiến hành gán vào biến, nếu hok thì xuất lỗi.
Hàm filter_input ở đây có thể dùng INPUT_GET thay cho INPUT_POST nếu me
thod của bạn là GET.
*/
?>
```

Đoạn code trên là khá dài, chúng ta rút ngắn lại như sau :

PHP Code:

```
$array = array('name'=>array('filter'=>FILTER_SANITIZE_STR
ING),
                'age' => array('filter'=>FILTER_VAL
IDATE_INT,
                                array('min_r
ange'=>18, 'max_range' =>80)),
                'email' =>array('filter'=>FILTER_VA
```

```

LIDATE_EMAIL) );
$result = filter_input_array(INPUT_POST, $array);
if($result['name'] && $result['age'] && $result['email']){
    extract($result); //Hàm này cắt mảng không tuần tự,
    tên của khóa sẽ là biến chứa giá trị khóa đó
}
var_dump($name);
var_dump($age);
var_dump($email);

```

Chúng ta có 1 loại filter đặc biệt là FILTER_CALLBACK .

PHP Code:

```

<?php
    $array_test =array('name'=>'Dang Hai Phung' , 'age' =>'18'
, 'email'=>'onlinemeanmissingyou@yahoo.com');
    $array = array('name'=>array('filter'=>FILTER_CALLBACK, 'op
tions'=>'strtoupper'),
                                'age' => array('filter'=>FILTER_VAL
IDATE_INT,
                                array('min_r
ange'=>18, 'max_range' =>80)),
                                'email' =>array('filter'=>FILTER_VA
LIDATE_EMAIL) );
    //Hàm strtoupper chuyển chuỗi thành chữ IN HOA
    $result = filter_var_array($array_test, $array);
    if(filter_var_array($array_test, $array)){
        extract($result);
    }
    var_dump($name);
    var_dump($age);
    var_dump($email);
?>

```

Các bạn thử trả lời các câu hỏi sau nhé :

- 1/ Filter là gì ? Tại sao phải dùng filter ?**
- 2/ Để dùng Filter cần nhớ các hàm nào ?**
- 3/ Kiểu lọc có mấy loại ? Trong mỗi loại thì có các loại FILTER_ nào ?**
- 4/ Cách sử dụng các hàm trong filter .**

Viết Class Database có chống SQL Injection

Bữa nay lần mò nghiên cứu về phòng chống lỗi SQL Injector. Nhân tiện mình viết 1 class Database để sử dụng lại cho tiện, cùng với mục đích lưu trữ kiến thức nên post lên đây cho mọi người cùng tham khảo, cũng không biết là viết như vậy đã hoàn chỉnh chưa. Nếu ai có cách viết khác hay hơn thì post lên chia sẻ nhé :).

PHP Code:

```
<?php
class DB_MySQL{
    private $db_host = "localhost"; //Khai báo thông tin host database
    private $db_user = "root"; //Khai báo thông tin username database
    private $db_pass = ""; //Khai báo thông tin password database
    private $db_name = "test"; // Khai báo thông tin name database
    private $connection;
    private $result = null; //Thuộc tính result trả về kết quả của query.

    private $magic_quotes_active;
    private $real_escape_string_exists;

    public function __construct(){
        $this->
        >openConnect(); // Tự động mở kết nối CSDL khi gọi đến class
        $this->magic_quotes_active = get_magic_quotes_gpc();
        $this->
        >real_escape_string_exists = function_exists( "mysql_real_escape_string" );
    }

    //Mở kết nối CSDL
    public function openConnect(){
        $this->connection = mysql_connect($this->db_host, $this->
        >db_user, $this->db_pass);
        if (!$this->connection){
            die("Database connection failed: " . mysql_error());
        }
        else{
            $db_select = mysql_select_db($this->db_name, $this->
        >connection);
            if(!$db_select){
                die("Database selection failed: " . mysql_error());
            }
        }
    }

    //Đóng kết nối CSDL
    public function closeConnect(){
        if ($this->connection)
        {
            mysql_close($this->connection);
            unset($this->connection);
        }
    }

    //Phương thức chống SQL Injection
    public function sqlQuote( $value ){
        //Kiểm tra xem version PHP bạn sử dụng có hiệu hàm mysql_real_e
        scape_string() hay ko

        if ($this->real_escape_string_exists) {
            //Trường hợp sử dụng PHP v4.3.0 trở lên
            //PHP hiệu hàm mysql_real_escape_string()
```

```

        if( $this->magic_quotes_active ) {
            //Trong trường hợp PHP đã hỗ trợ hàm get_magic_quotes_g
pc()
            //Ta sử dụng hàm stripslashes để bỏ qua các dấu slashes
            $value = stripslashes( $value );
        }
        $value = mysql_real_escape_string( $value );
    }
    else {
        //Trường hợp dùng cho các version PHP dưới 4.3.0
        //PHP không hiểu hàm mysql_real_escape_string()

        if( !$this->magic_quotes_active ){
            //Trong trường hợp PHP không hỗ trợ hàm get_magic_quote
s_gpc()
            //Ta sử dụng hàm addslashes để thêm các dấu slashes vào
            giá trị
            $value = addslashes( $value );
        }
        // Nếu hàm get_magic_quotes_gpc() đã active có nghĩa là các
        dấu slashes đã tồn tại rồi
    }
    return $value;
}

//Chạy câu truy vấn query
public function query($sql){
    $this->result = mysql_query($sql, $this->connection);
    if (!$this->result){
        $output = "Database query failed: " . mysql_error() . "<br
/><br />";
        die($output);
    }
    return $this->result;
}

//Lấy số record dữ liệu mảng trong CSDL thông qua câu truy vấn quer
y
public function fetch_array(){
    if ($this->result){
        $rows = array();
        $rows = mysql_fetch_array($this->result);
    }
    return $rows;
}

//Đếm số record trong CSDL thông qua câu truy vấn query
public function num_row(){
    if($this->result){
        $num = null;
        $num = mysql_num_rows($this->result);
    }
    return $num;
}
}
?>

```

Cách sử dụng:

PHP Code:

```
$username = $_POST['username'];
$password = $_POST['password'];
$db = new DB_MySQL;
$sql = "SELECT * FROM user WHERE username='". $db->sqlQuote($username)."' AND password='". $db->sqlQuote($password)."'";
$db->query($sql);
.....
$db->closeConnect();
```

Có thể chống SQL Injection 1 cách linh hoạt hơn bằng cách sử dụng sqlQuote để thiết lập các giá trị input trước khi đưa nó vào câu truy vấn sql.

Cách lấy rss và tin tức Từ VnExpress

Bài viết này hướng dẫn các bạn lấy tiêu đề,tóm tắt,nội dung và RSS thông qua rss của trang VnExpress.Còn rất nhiều hạn chế mạnh dạn post lên các pro góp ý nhé.
Trước tiên là hàm lấy tin tức (hàm này của thầy Huy ^^ nhưng mình thấy thầy lấy tin từ 1 bài thôi)

PHP Code:

```
function getdata($url, $start, $stop, $str_to_replace='', $str_replace=
'', $extra_data='')
{
    $fd = "";
    $start_pos = $end_pos = 0;
    $url = fopen($url, "r");//url bạn muốn lấy tin
    while(true) {
        if($end_pos > $start_pos) {
            $result = substr($fd, $start_pos, $end_pos-$start_pos);
            $result .= $stop;
            break;
        }
        $data = fread($url, 8192);
        if(strlen($data) == 0) break;
        $fd .= $data;
        if(!$start_pos) $start_pos = strpos($fd, $start);
    }
}
```

```

if($start_pos) $end_pos = strpos(substr($fd, $start_pos), $stop) + $start_pos;
}
fclose($url);
$info=str_replace($str_to_replace, $str_replace, $extra_data.$result);
return $info;
}

```

Các bạn mở bất kì file rss nào của VnExpress để thấy dc cấu trúc file XML có dạng (Lên mạng đọc cấu trúc xml mình thấy cũng giống với cấu trúc html ^^)

PHP Code:

```

<item>//root element (cái này coi như thẻ cha bên trong là các thẻ con)
    <title><![CDATA[ Ngân hàng trả lương cao nhất khối tổng công ty nhà n
ước ]]></title>    <description><![CDATA[ <a href="http://vnexpress.n
et/gl/ebank/thi-truong/2011/09/ngan-hang-tra-luong-cao-nhat-khoi-tong-
cong-ty-nha-
nuoc/"></a>Vượt qua các "quả bom thép" thuộc các ngành dầu khí, viễ
n thông, khối ngân hàng Nhà nước đang đứng vị trí đầu bảng về mức chi t
rả thu nhập cho cán bộ nhân viên với mức bình quân một tháng lên tới 18
triệu đồng. ]]></description>    <link>http://ebank.vnexpress.net/gl/eba
nk/thi-truong/2011/09/ngan-hang-tra-luong-cao-nhat-khoi-tong-cong-ty-
nha-
nuoc/</link>    <pubDate>Wed, 14 Sep 2011 17:01:02 GMT</pubDate>    </i
tem>

```

Tiếp đến là đọc XML DOM,với cấu trúc file rss như trên

PHP Code:

```

$dom=new DOMDocument('1.0','utf-8');//tao doi tuong dom
$dom-
>load("Trang rss muốn lấy");//muon lay rss tu trang nao thi ban khai ba
o day
    $items = $dom-
>getElementsByTagName("item");//lay cac element co tag name la item va
gan vao bien $items
    foreach($items as $item)//lap
    {
        $titles=$item-
>getElementsByTagName('title');//lay cac element co tag name la title v
a gan vao bien $titles
        $title=$titles-
>item(0);//lay ra gia tri dau tuiien trong array $titles
        $descriptions=$item->getElementsByTagName('description');
        $des=$descriptions->item(0);
        $links=$item->getElementsByTagName('link');
        $link=$links->item(0);
        $pubDates=$item->getElementsByTagName('pubDate');
        $pubDate=$pubDates->item(0);
    }

```

Và đây là cách sử dụng chỉ việc echo ra những gì bạn muốn lấy,lúc này có thể thêm tiêu đề,nội dung,tóm tắt,ngày gửi vào CSDL của bạn

PHP Code:

```

<table width="800" border="1" align=center bgcolor=#e5eef2>
    <tr>
        <td style="color:#FF0000; font-weight:bold; text-
decoration:none"><a href="<? echo $link-

```

```

<nodeValue ;?>" /><? echo $title->nodeValue ?></td>
</tr>
<tr>
<td>
    <?php
        echo $pubDate->nodeValue;
        echo $dec->nodeValue;
    ?>
</td>
</tr>
<tr>
<td><font face=arial size=2>
    <?php
        $url=$link->nodeValue;
        $start="<div id=\"content\">";
        $end="<div class=\"tag-parent\">";
        $img_search= "/Files";
        $img_show="http://vnexpress.net/Files";
        $extra='<META HTTP-EQUIV="Content-Type" Content="text-
html; charset=UTF-
8"><link href="/Resource/Common/default.css" rel="stylesheet" type="tex
t/css">';
        echo getdata($url,$start,$end,$img_search,$img_show,$extra);
    ?>
</font></td>
</tr>
</table>

```

Như vậy cứ sau 1 vòng lặp thì ta sẽ lấy được link của <link></link> Trong file xml và gán link đó vào biến \$url, như vậy sẽ lấy dc tất cả link chứ không chỉ 1 link

Đây là toàn bộ code của bài này

PHP Code:

```

<?

function getdata($url, $start, $stop, $str_to_replace='', $str_replace=
'', $extra_data='')
{
    $fd = "";
    $start_pos = $end_pos = 0;
    $url = fopen($url, "r");
    while(true) {
        if($end_pos > $start_pos) {
            $result = substr($fd, $start_pos, $end_pos-$start_pos);
            $result .= $stop;
            break;
        }
        $data = fread($url, 8192);
        if(strlen($data) == 0) break;
        $fd .= $data;
        if(!$start_pos) $start_pos = strpos($fd, $start);
        if($start_pos) $end_pos = strpos(substr($fd, $start_pos), $stop) + $sta
rt_pos;
    }
    fclose($url);

    $info=str_replace($str_to_replace, $str_replace, $extra_data.$result);
}

```

```

return $info;
}
?>
<ul style='list-style:none;'>
  <li><a href='?mod=xa-hoi'>Xã Hội</a></li>
  <li><a href='?mod=the-gioi'>Thế Giới</a></li>
  <li><a href='?mod=kinh-doanh'>Kinh Doanh</a></li>
  <li><a href='?mod=van-hoa'>Văn Hóa</a></li>
  <li><a href='?mod=the-thao'>Thể Thao</a></li>
</ul>
<?
  $file='http://vnexpress.net/rss/gl/xa-hoi.rss';
  switch($_REQUEST['mod'])
  {
    case "xa-hoi":
      $file='http://vnexpress.net/rss/gl/xa-hoi.rss';
      break;
    case "the-gioi":
      $file='http://vnexpress.net/rss/gl/the-gioi.rss';
      break;
    case 'kinh-doanh':
      $file='http://vnexpress.net/rss/gl/kinh-doanh.rss';
      break;
    case 'van-hoa':
      $file="http://vnexpress.net/rss/gl/van-hoa.rss";
      break;
    case 'the-thao':
      $file='http://vnexpress.net/rss/gl/the-thao.rss';
      break;
    default:
      $file='http://vnexpress.net/rss/gl/xa-hoi.rss';
      break;
  }
  $dom=new DOMDocument('1.0','utf-8');//tao doi tuong dom
  $dom->load($file);//muon lay rss tu trang nao thi ban khai bao day
  $items = $dom-
>getElementsByTagName("item");//lay cac element co tag name la item va
gan vao bien $items
  foreach($items as $item)//lap
  {
    $titles=$item-
>getElementsByTagName('title');//lay cac element co tag name la title v
a gan vao bien $titles
    $title=$titles-
>item(0);//lay ra gia tri dau tuiien trong array $titles
    $descriptions=$item->getElementsByTagName('description');
    $des=$descriptions->item(0);
    $links=$item->getElementsByTagName('link');
    $link=$links->item(0);
    $pubDates=$item->getElementsByTagName('pubDate');
    $pubDate=$pubDates->item(0);

  }
?>

<table width="800" border="1" align=center bgcolor=#e5eef2>

```

```

        <tr>
            <td style="color:#FF0000; font-weight:bold; text-
decoration:none"><a href="<? echo $link-
>nodeValue ;?>" /><? echo $title->nodeValue ?></td>
        </tr>
        <tr>
            <td>
                <?php
                    echo $pubDate->nodeValue;
                    echo $dec->nodeValue;
                ?>
            </td>
        </tr>
        <tr>
            <td><font face=arial size=2>
                <?php
                    $url=$link->nodeValue;
                    $start="<div id=\"content\">";
                    $end="<div class=\"tag-parent\">";
                    $img_search= "/Files";
                    $img_show="http://vnexpress.net/Files";
                    $extra='<META HTTP-EQUIV="Content-Type" Content="text-
html; charset=UTF-
8"><link href="/Resource/Common/default.css" rel="stylesheet" type="tex
t/css">';
                    echo getdata($url,$start,$end,$img_search,$img_show,$extra);
                ?>
            </font></td>
        </tr>
    </table>

    <?
    }
    ?>

```

**Không hiểu sao khi làm localhost thì chạy ngon va upload lên host thì không có gì
^^.Bác nào biết tại sao không,Chắc do host rồi lên không demo dc,chịu khó tạo file
php chạy trên localhost nha
Ai có cách lấy tin nào hay post lên anh em tham khảo nhé**

Cách dùng hàm ereg, preg trong php

Để trả lời cho câu hỏi về chức năng của hàm ereg, mình nêu ra một số tình huống:

+ Bạn có một chuỗi dữ liệu: \$xxx, bạn cần kiểm tra xem chuỗi đó có phải là định dạng: 4 con số đứng liền nhau, rồi đến dấu -, rồi đến 2 chữ số, rồi đến dấu -, rồi đến 2 chữ số nữa.

+ Bạn có một chuỗi dữ liệu: \$xxx, bạn cần tìm trong chuỗi \$xxx, chỗ nào có định dạng là: dấu <, rồi các kí tự, rồi đến dấu >. Sẽ thay thế bằng chữ yyy. Ví dụ: \$xxx= "<a>noi dung gi do <p> noi dung tiep theo"; sẽ được biến đổi thành: "yyy noi dung gi do yyy noi dung tiep theo".

Rõ ràng ở 2 tình huống trên thì các hàm string của chúng ta không xử lý được. Bởi vì các chuỗi chúng ta cần so sánh hoặc cần tìm là không xác định, ta chỉ biết định dạng của nó thôi.

Do đó php hỗ trợ cho ta 2 nhóm hàm là ereg... và preg... để xử lý các tình huống trên.

Mình đề nghị bạn nên dùng nhóm hàm preg... để thay thế cho nhóm hàm ereg.... Vì:

+ ereg... không còn hỗ trợ ở các phiên bản sau (ereg... has been DEPRECATED as of PHP 5.3.0. Relying on this feature is highly discouraged.)

+ preg... chạy nhanh hơn ereg... (preg... which uses a Perl-compatible regular expression syntax, is often a faster alternative to ereg().)

B. Cách sử dụng:

+ Vì những lý do ở trên, mình đề nghị bạn nên chuyển sang dùng nhóm hàm preg... luôn.

+ Sau đây là một số hàm thuộc nhóm preg....:

preg_match() – Perform a regular expression match (so sánh)

preg_replace() – Perform a regular expression search and replace (tìm kiếm và thay thế)

preg_split() – Split string by a regular expression (tìm kiếm và cắt bỏ)

preg_last_error() – Returns the error code of the last PCRE regex execution

preg_match_all() – Perform a global regular expression match

Tiếp theo, mình sẽ bàn thêm cách viết các biểu thức định dạng như thế nào? Tức là: nếu viết bằng văn thì là: "4 con số đứng liền nhau, rồi đến dấu -, rồi đến 2 chữ số, rồi đến dấu -, rồi đến 2 chữ số nữa" nhưng viết cho php hiểu thì viết như thế nào?

C. Một số lưu ý khi viết chuỗi định dạng:

+ Dùng cặp dấu ./ để bắt đầu và kết thúc chuỗi định dạng.

```
<?php
if (preg_match("/php/", "PHP is the web scripting language of choice.")) {
echo "A match was found.";
} else {
echo "A match was not found.";
}
?>
```

==> Kiểm tra xem có tồn tại chuỗi "php" không? Kết quả là không. Vì trong chuỗi mẹ

chỉ có chữ PHP (viết hoa) thôi.

+ Để chỉ định không phân biệt chữ hoa hay chữ thường ta dùng kí hiệu i.

```
<?php
if (preg_match("/php/i", "PHP is the web scripting language of choice.")) {
echo "A match was found.";
} else {
echo "A match was not found.";
}
?>
```

==> Kết quả là có tồn tại.

+ Dấu ^ để qui định chuỗi định dạng nằm ở vị trí đầu tiên của chuỗi mẹ.

```
<?php
if (preg_match("/^php/i", "Content PHP is the web scripting language of choice.")) {
echo "A match was found.";
} else {
echo "A match was not found.";
}
?>
```

==> Kết quả là ko tìm thấy. Vì chữ php không nằm ở vị trí đầu tiên

+ Dấu \$ qui định chuỗi định dạng nằm ở vị trí cuối cùng của chuỗi mẹ.

```
<?php
if (preg_match("/php$/i", "PHP is the web scripting language of choice. And I choice
PHP")) {
echo "A match was found.";
} else {
echo "A match was not found.";
}
?>
```

==> Kết quả là tìm thấy. vì có chữ PHP ở cuối chuỗi mẹ.

+ Dấu \s để qui định khoảng trắng

```
<?php
if (preg_match("/php(\s)pro/i", "PHP pro is the web scripting language of choice")) {
echo "A match was found.";
} else {
echo "A match was not found.";
}
?>
```

==> Tìm chuỗi php pro (có khoảng trắng ở giữa chữ php và chữ pro). Kết quả là có tồn tại

+ Dấu ? để qui định: không tồn tại, hoặc nếu tồn tại thì chỉ có 1 thôi

```
<?php
if (preg_match("/php(\s)?pro/i", "PHPpro is the web scripting language of choice. And I
choice PHP")) {
echo "A match was found.";
} else {
echo "A match was not found.";
}
```

```
}  
?>
```

==> Tìm chuỗi phppro hoặc php pro. Kết quả là tồn tại.

+ Dấu * để qui định: không tồn tại, hoặc nếu có thì có thể có nhiều

```
<?php  
if (preg_match("/php(\s)*pro/i", "PHP pro is the web scripting language of choice. And I  
choice PHP")) {  
echo "A match was found.";  
} else {  
echo "A match was not found.";  
}  
?>
```

==> Tìm chuỗi phppro hoặc php pro (ở giữa có thể có nhiều khoảng trắng). Kết quả là tồn tại.

+ Dấu + để qui định: phải tồn tại từ 1 trở lên.

```
<?php  
if (preg_match("/php(\s)+pro/i", "PHPpro is the web scripting language of choice. And I  
choice PHP")) {  
echo "A match was found.";  
} else {  
echo "A match was not found.";  
}  
?>
```

==> Tìm chuỗi php pro (ở giữa phải có từ 1 khoảng trắng trở lên). Kết quả là không tồn tại.

+ Cặp dấu [] để liệt kê

```
<?php  
if (preg_match("/[_.]php/i", "_PHP is the web scripting language of choice")) {  
echo "A match was found.";  
} else {  
echo "A match was not found.";  
}  
?>
```

==> Tìm chuỗi "_php" hoặc ",php". Kết quả là tồn tại

+ A-Z, a-z, A-z qui định một kí tự viết hoa, viết thường, tất cả. Phải đi kèm với dấu []

```
<?php  
if (preg_match("/[a-z]php/i", "aPHP is the web scripting language of choice")) {  
echo "A match was found.";  
} else {  
echo "A match was not found.";  
}  
?>
```

==> Kết quả là tồn tại

+ 0-9 qui định một kí số

```
<?php  
if (preg_match("/[0-9]php/i", "_PHP is the web scripting language of choice")) {
```

```

echo "A match was found.";
} else {
echo "A match was not found.";
}
?>
==> Kết quả ko tồn tại
+ [A-z0-9]: một kí tự hoặc một kí số
<?php
if (preg_match("/[A-z0-9]php/i", "aPHP is the web scripting language of choice")) {
echo "A match was found.";
} else {
echo "A match was not found.";
}
?>
==> Kết quả là tồn tại
+ {} qui định số lượng
<?php
if (preg_match("/[a-z]{2,5}php/i", "aPHP is the web scripting language of choice")) {
echo "A match was found.";
} else {
echo "A match was not found.";
}
?>
==> Tìm chuỗi mà có từ 2 đến 5 kí tự rồi đến chữ php. Kết quả ko tồn tại. Vì aPHP phía
trước chữ PHP chỉ có một kí tự

```

Khi dùng ajax để khi nhấn enter sẽ tự động gọi hàm giống khi dùng form

Chào các bạn, khi mình dùng ajax (login ajax hay tìm kiếm bằng ajax) thì nút button bạn sẽ để sự kiện onclick vậy thì làm sao mà khi nhấn enter nó sẽ tự biết gọi hàm và tìm kiếm hay login giống như submit form ?

Chắc nhiều bạn đã biết nhưng mình vẫn giới thiệu cho bạn nào chưa biết.

thì dùng như thế này:

Vẫn dùng form bạn ạh nhưng gọi được hàm ajax :
bạn viết form như mình thường nhưng form bạn chỉ cần để onsubmit="return false;"

HTML Code:

```
<form onsubmit="return false;">
```

như bạn thấy mình để sự kiện onsubmit= return false

HTML Code:

```
<input type="image" src="images/login_button.jpg" onclick="ham();" />
```

vậy là nhập giá trị vào textbox nhấn enter chứ không cần phải click là nó gọi hàm được các bạn àh

Ví dụ

HTML Code:

```
<form onsubmit="return false;">
```

```
    <input type="text" id="key" /><br />
```

```
    <input type="image" src="abc.jpg" onClick="tim();" />
```

```
</form>
```

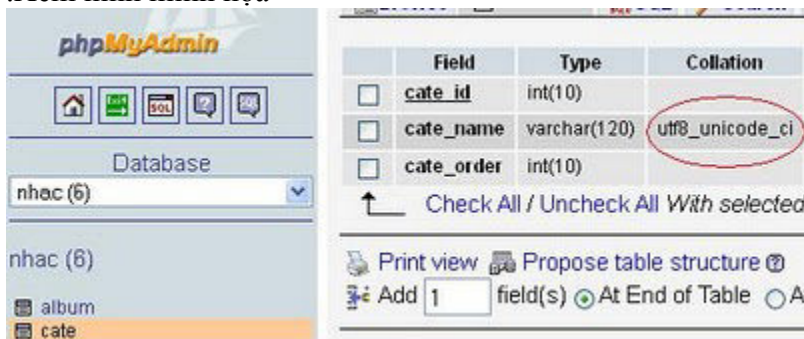
[TUT] Add dữ liệu từ form vào database hiển thị tiếng việt có dấu.

Tôi nay viết cái này hướng dẫn bạn add dữ liệu từ form vào database (phpAdmin) và hiển thị đc tiếng việt có dấu trong database (phpAdmin) của mysql.

-Bước 1: Tạo 1 database và các table với Collation là utf8_unicode_ci. Nếu bạn đã có sẵn database thì chuyển Collation của database về utf8_unicode_ci luôn nhé.

Ở đây mình lấy ví dụ database của mình là nhạc và table là cate

.Xem hình minh họa



-Bước 2: Trong file .php sau khi bạn khai báo câu lệnh kết nối đến cơ sở dữ liệu thì bạn thêm câu lệnh này phía dưới nó mysql_set_charset('utf8',\$conn);

Xem code minh họa:

PHP Code:

```
<?php
$conn=@mysql_connect("localhost","root","root");
@mysql_set_charset('utf8',$conn);
@mysql_select_db("nhac",$conn);
?>
```

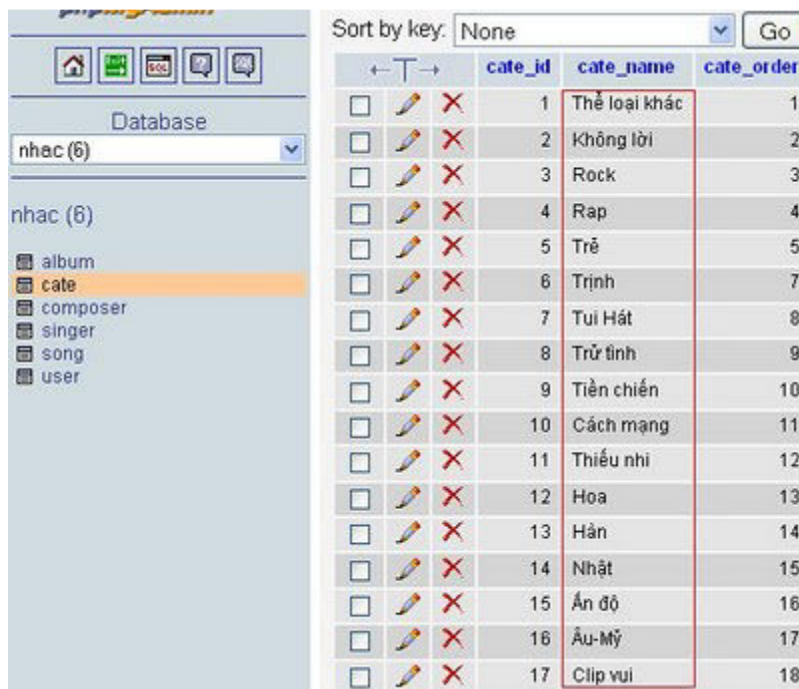
-Bước 3: Giờ thì bạn vào form add thử 1 record mới xem nào. Ở đây Tân lấy ví dụ là table cate nhé:

Thêm thể loại

Tên thể loại:

Số thứ tự:

-Sau khi thêm thể loại thành công thì chạy vào localhost xem thử cái nào



	cate_id	cate_name	cate_order
<input type="checkbox"/>	1	Thể loại khác	1
<input type="checkbox"/>	2	Không lời	2
<input type="checkbox"/>	3	Rock	3
<input type="checkbox"/>	4	Rap	4
<input type="checkbox"/>	5	Trẻ	5
<input type="checkbox"/>	6	Trịnh	7
<input type="checkbox"/>	7	Tui Hát	8
<input type="checkbox"/>	8	Trữ tình	9
<input type="checkbox"/>	9	Tiền chiến	10
<input type="checkbox"/>	10	Cách mạng	11
<input type="checkbox"/>	11	Thiếu nhi	12
<input type="checkbox"/>	12	Hoa	13
<input type="checkbox"/>	13	Hàn	14
<input type="checkbox"/>	14	Nhật	15
<input type="checkbox"/>	15	Ấn độ	16
<input type="checkbox"/>	16	Âu-Mỹ	17
<input type="checkbox"/>	17	Clip vui	18

-Còn đây là dữ liệu đc lấy từ database nhac đổ ra bảng trên web nè

STT	Thể loại	Số thứ tự	Sửa	Xóa
1	Thể loại khác	1	[+]	[-]
2	Không lời	2	[+]	[-]
3	Rock	3	[+]	[-]
4	Rap	4	[+]	[-]
5	Trẻ	5	[+]	[-]
6	Trình	7	[+]	[-]
7	Tui Hát	8	[+]	[-]
8	Trữ tình	9	[+]	[-]
9	Tiền chiến	10	[+]	[-]
10	Cách mạng	11	[+]	[-]
11	Thiếu nhi	12	[+]	[-]
12	Hoa	13	[+]	[-]
13	Hàn	14	[+]	[-]
14	Nhật	15	[+]	[-]
15	Ấn độ	16	[+]	[-]
16	Âu-Mỹ	17	[+]	[-]
17	Clip vui	18	[+]	[-]

Có gì thắc mắc cứ comment hén ! 

Tản mạn về ECHO

Chắc các bạn chẳng bao giờ đọc Document về echo đâu nhỉ? Nhưng hôm nay mình e rằng các bạn sẽ phải khá bất ngờ, vì tại sao mình lại viết bài này, thật sự thì mình cũng hơi bất ngờ khi vô tình đọc phải document về nó trên php.net

Chức năng: Xuất ra 1 hoặc nhiều chuỗi

Các ứng dụng về nó

1. Xuất ra 1 chuỗi bình thường (Xưa quá rồi)

PHP Code:

```
echo 'Hello QHO Solution';
```

2. Xuất ra 1 chuỗi có xuống dòng (Cũng xưa rồi)

PHP Code:

```
echo 'Hello\nQHO\nSolution';
```

3. Xuất ra 1 chuỗi có kèm dấu nháy kép (Cái này nên lưu ý vì các bạn hay sử dụng)

PHP Code:

```
// Cách 1
```

```
echo 'Hello "QHO Solution"';
```

```
// Cách 2
echo "Hello \"QHO Solution\"";

// ----- //
Rõ ràng cách 1 hay hơn cách 2 đúng ko nào?
```

4. Sự khác nhau giữa nhảy đơn và nhảy kép

PHP Code:

```
$var = 'QHO Solution';
echo 'Hello $var'; // Kết quả là Hello $var
echo "Hello $var"; // Kết quả là Hello QHO Solution

// Còn 1 cách nữa là nối chuỗi và biến, xưa quá rồi, mình không nói thêm nhé
```

5. Sự thay thế của Multiple echo

PHP Code:

```
echo '<div>';
echo "<p>$a</p>";
echo '<input type="checkbox" name="abc" value="'. $b. '" />';
echo '</div>';

// Đoạn trên đc rút gọn lại thành

echo <<<END
<div>
    <p>$a</p>
    <input type="checkbox" name="abc" value="$b" />
</div>
END;

// Rõ ràng cách viết dưới gọn hơn cách trên, đồng thời code HTML được tab vào dễ nhìn hơn
```

6. Echo có điều kiện (áp dụng IF ELSE SHORT HAND)

PHP Code:

```
$a = 'Solution';
echo ($a == 'Solution')?'QHO Solution':'QHONLINE';
// Kết quả là QHO Solution
```

Trong những trường hợp trên, mình 'khoái' nhất là cái thứ 5 ^^, thật bất ngờ khi tình cờ đọc thấy nó, 1 chút chia sẻ cho những bạn chưa biết

Code xem thư mục con .

Hi mọi ng ,mới học được cái này nên share cho mọi người nè :). Share chủ yếu để mình nhớ kiến thức lâu hơn, còn chắc mọi ng cũng biết hết rồi :D

Code xem thư mục con bằng php :

Hiện thị thư mục con trước : (bạn có thể thay tên " phpMyAdmin " thành tên khác)

test_1.php

PHP Code:

```
<table width=100% bgcolor='#cccccc'>
<?php
    $size_1=GetImageSize('images/images.jpg');
    var_dump($size_1);
    $size_1[0]=$size_1[0]/5;
    $size_1[1]=$size_1[1]/5;
    $size_2=GetImageSize('images/name.jpg');
    $size_2[0]=$size_2[0]/5;
    $size_2[1]=$size_2[1]/5;
    $dir=opendir('phpMyAdmin');
    while (($file=readdir($dir))!==false)
    {
        $file_or_folder="phpMyAdmin/$file";
        echo "<tr><td>";
            if(is_dir($file_or_folder)  && $file!='.' && $file
e!='..' )
            {
                $target="test_2.php?name=".$file;
                echo "<a href=\"".$target.\"\"><img src=\"images/i
images.jpg\" width=\"".$size_1[0].\"\" height=\"".$size_1[1].\"\"/>$file</
a></td>";
            }
            if( !is_dir($file_or_folder)  && $file!='.' && $
file!='..')
                echo " <img src=\"images/name.jpg\" width=\"".$siz
e_2[0].\"\" height=\"".$size_2[1].\"\"/>$file</td>";
            }
        closedir($dir);
    }
?>
</table>
```

test_2.php

PHP Code:

```
<table width=100%>
<?php
    $size_1=GetImageSize('images/images.jpg');
    $size_1[0]=$size_1[0]/5;
    $size_1[1]=$size_1[1]/5;
    $size_2=GetImageSize('images/name.jpg');
    $size_2[0]=$size_2[0]/5;
    $size_2[1]=$size_2[1]/5;
    $name=$_GET['name'];

    if(strpos($name,'phpMyAdmin')===false)
    {
        $file_0="folder/$name/";
        $dir=opendir($file_0);
    }
    else
```

```

{
    $file_0="$name/";
    $dir=opendir($name);
}

while (($file=readdir($dir))!==false)
{
    $file_or_folder="$file_0/$file";
    echo "<tr><td>";
        if(is_dir($file_or_folder) && $file!='.' &&
&& $file!='..')
        {
            $temp=$file;
            $file=$file_0.$file;
            $target="test_2.php?name=".$file;
            echo "<a href=\"".$target.\"><img src=\"image
s/images.jpg\" width=\"".$size_1[0].\" height=\"".$size_1[1].\"/>$tem
p</a></td>";
        }

        if(!is_dir($file_or_folder) && $file!='.' &&
$file!='..')
            echo " <img src=\"images/name.jpg\" width=\"
".$size_2[0].\" height=\"".$size_2[1].\"/>$file</td>";
    }
    closedir($dir);
?>
</table>

```

Ở trên dùng : opendir() : mở thư mục, readdir() : đọc thư mục, is_dir(): kiểm tra xem thư mục đó là folder hay file (folder return true, not : return false) .
 Trong 1 folder nó có cái foldername && filename là '.' với '..' . Mình hok biết 2 đũa này ở đâu ra nhưng mà chặn nó ngay folder thì nó lại xuất hiện dưới dạng file và ngược lại nên mình chặn 2 cái luôn (if (is_dir... && \$file!='.' && \$file!='..')) .

Test 1.php :

Mình download hình về nó to nên mình dùng hàm GetImageSize() để lấy width (size[0]) và height(size[1]) rồi chia 5 >< .

Lưu ý là để kiểm tra có phải folder hay hok, mình phải chèn vào trước tên của file or folder tên thư mục cha để thành đường dẫn đầy đủ . Nên dùng biến \$file_or_folder để đặt tiền tố .

Test 2.php:

Nguyên lý của test_2 .php là nó chạy lặp lại, từ test_2.php sẽ chạy típ test_2.php thông qua phương thức \$_GET . Nên mình phải kiểm xem, \$name ở đây đã có tiền tố chưa (nếu có rồi tức là test_2.php đã bắt đầu lặp) . Nếu chưa thì phải chèn tiền tố vào rồi mới opendir().

Cơ cấu lặp : lấy \$temp giữ nội dung \$file để lát xuất ra màn hình .

Sau đó \$file sẽ dc chèn tiền tố \$file_0 vào . Rồi gán \$target (đường trở link) = \$file

Rồi test_2.php lặp lại, lúc này \$name sẽ là \$file sau đó \$file_0 dc gán = \$name thêm dấu '/' . Rồi chạy tiếp cho folder kế tiếp cứ thế đi hết sơ đồ cây :) .
Bạn test bài này bằng cách tạo 1 filename.php trong www rồi bỏ folder vào www sẽ dc :)
Chào mọi người . Có code nào ngắn hơn thì cho mình học hỏi nhé :)

STRING trong PHP

[PART1]

I/String là gì

Biến trong PHP có thể dùng để lưu trữ và thao tác với dạng chữ . Chuỗi được chứa phải nằm trong nháy đơn (' ') hay nháy đôi (" ") .

VD: \$string = ' Hello world ' ;

\$string = " Hello world " ;

Điểm khác nhau giữa nháy đơn và nháy đôi:

Nháy đơn có 1 nhiệm vụ duy nhất : chứa những gì ở bên trong nó . Nháy đơn có thể bao gồm nháy đôi ở bên trong mà không bị lỗi

Nháy đôi có thêm 1 nhiệm vụ nữa : nếu ở trong nó có biến chứa thông tin khác hay kí tự có nghĩa như \n , thì tự động thay thế biến bằng thông tin đó hay làm nhiệm vụ của \n .
Tạm gọi đây là 'NỘI SUY' . Nháy đôi có thể bao gồm nháy đơn và biến mà không bị lỗi

Vậy, cái nào làm nhiều hơn thì cái đó sẽ chậm hơn. Nên chỉ **khi cần thiết** chúng ta mới dùng nháy đôi .

Lưu ý : biến môi trường không thể nội suy được . Như \$_POST, \$_GET, Chúng ta phải dùng 1 biến khác để chứa nó rồi nội suy biến vừa tạo thì được .

Nối chuỗi:

Dùng toán tử " . " để nối chuỗi

VD:

PHP Code:

```
$a = 'Your name is : ' ;  
$b = 'Jack';  
$c = $a.$b;  
echo $c; //Xuất 'Your name is : Jack' .
```

Con trỏ chuỗi : Chuỗi được thao tác dựa trên con trỏ bên trong chuỗi .

VD:

\$a = 'This is a letter';

Chuỗi trên có 16 kí tự tính luôn khoảng trắng, vậy vị trí có thể của con trỏ là từ 0 -> 15 .
Và nó đại diện cho kí tự ngay sau nó . Hiểu về con trỏ chuỗi sẽ dễ dàng làm việc với 1 số hàm cắt chuỗi hơn .

II/Một số hàm quan trọng để cắt chuỗi hay tìm trong chuỗi:

Trong nhiều trường hợp chúng ta cần cắt chuỗi . 1 số hàm giúp ta làm điều đó là :

substr(string, start, length) :

Gọi kí tự đầu chuỗi được cắt ra là X , kí tự cuối chuỗi được cắt ra là Y

Nếu giá trị của start:

- Dương: X là kí tự thứ start của chuỗi ban đầu
- Âm: X là ký tự cách phía phải của chuỗi ban đầu giá trị tuyệt đối 'start' đơn vị
- 0 : từ vị trí đầu chuỗi

Nếu giá trị của length :

- Dương : chiều dài của chuỗi được cắt
- Âm : Y cách phía phải một giá trị tuyệt đối 'length' đơn vị
- 0 : Hết cắt gì hết
- Không chỉ ra length : cắt tới hết chuỗi

VD:

PHP Code:

```
$a = ' String to test function '; //25 ký tự
    $b_1 = substr($a,3,6);
    var_dump($b_1);
//Cắt ra 'ring t'
    $b_2 = substr($a,3,-6);
    var_dump($b_2);
//Cắt ra 'ring to test fun'
    $b_3 = substr($a,3,0);
    var_dump($b_3);
//Cắt ra rỗng ''
$b_4 = substr($a,-3,6);
    var_dump($b_4);
//Cắt ra 'on '
    //1 số trường hợp đặc biệt:
    $b_5 = substr($a, -3,-6);
    var_dump($b_5);
//Cắt từ vị trí 0, kết thúc ở vị trí c <<<Ra rỗng ''
    $b_6 = substr($a,-3,6);
    var_dump($b_6);
//Cắt từ vị trí 0, cắt 6 kí tự <<< ra 'on ' vì maximum cắt được chỉ có 3 thôi
```

strstr(\$string, 'string_to_search') : trả về một chuỗi bắt đầu từ 'string_to_search' đầu tiên trong \$string nếu có, false nếu hok có .

VD:

PHP Code:

```
$string = ' This IS a test of string ';
    $result = strstr($string, 'IS');
    var_dump($result);
//Trả về 'IS a test of string'
```

stristr(): y chang strstr() nhưng hok phân biệt hoa-thường

strpos(string, 'string_to_search' ,int start) : trả về VỊ TRÍ của 'string_to_search' đầu tiên sau vị trí 'start' (nếu có sử dụng start) trong \$string , hay false nếu 'string_to_search' hok có trong \$string

VD:

PHP Code:

```
$string = ' This IS a test of string ';
$result = strpos($string, 'is', 4);
```

```
var_dump($result);
//Trả về FALSE vì hàm strpos phân biệt hoa thường
$result = strpos($string, 'is');
var_dump($result);
//Trả về '3'
```

Strpos(): y chang strpos() nhưng hok phân biệt hoa - thường

Strrpos(\$string, 'string_to_search', int start) : trả về chuỗi bắt đầu từ VỊ TRÍ của 'string_to_search' cuối cùng (sau start nếu có) trong \$string nếu tìm thấy, false nếu hok tìm thấy[/PHP]

VD:

PHP Code:

```
$string = ' This is a test of string ';
$result = strrpos($string, 'is');
//Trả về '6' thay vì '3'
```

Stripos(string, 'string_to_search', int start) : giống như **strrpos()** nhưng hok phân biệt hoa thường

Str_replace(find,replace,string,count): tìm và thay thế trong chuỗi, biến count dùng để đếm số lần thay thế . Hàm này phân biệt hoa thường

Một số lưu ý :

1. Nếu string là 1 mảng: tìm và thay thế của từng phần tử ở trong mảng và trả về kiểu dữ liệu array
2. Nếu find là 1 mảng : tìm và thay thế trong \$string các giá trị trong mảng find .
3. Nếu find và replace cùng là mảng và mảng replace ít phần tử hơn mảng string : tìm và thay thế lần lượt, phần dư thì trả về rỗng

VD:

PHP Code:

```
$string = 'This is a test of replacing in string' ;
$result = str_replace('is', 'IS', $string, $count);
var_dump($result);
// Trả về 'ThIS IS a test of replacing in string';
var_dump($count); //Trả về 2
$string_array = array('a','b','c','a','c');
$array_test_1 = str_replace('a', 'replaced',$string_array);
var_dump($array_test_1);
// $array_test_1 bây giờ là array('replaced','b','c','replaced','c');
/*(1) */$find_test_array = array('a','c');
$string_array = array('a','b','c','a','c');
$array_test_2 = str_replace($find_test_array, 'abc', $string_array);
var_dump($array_test_2);
//Code ở trên có nghĩa : tìm 'a' trong $string_array rồi thay thế bằng 'abc', sau đó tìm tiếp 'c' trong mảng trả về rồi lại thay thế bằng 'abc'
// $array_test_(2) bây giờ là array('ababc','b','abc','ababc','abc')
$string_array = array('a','b','c','a','c','e');
$find_array =array ('a' , 'c', 'd', 'e');
$replace_array = array('r_a', 'r_c', 'r_d');

/*(2)*/ $array_test2 = str_replace($find_array, $replace_array, $string_array);
// Trả về array('r_a','b','r_c', 'r_a', 'r_c', '')
```

str_ireplace() : giống **str_replace()** nhưng hok phân biệt hoa- thường

substr_replace(string, replace , start, length) :

Tính chất của start thì như ở trên hàm ...
Nếu length:

- Dương: độ dài chuỗi sẽ bị thế
- Âm : độ dài từ vị trí cuối cùng được thay thế cho đến cuối chuỗi
- 0 : thêm vào thay vì thay thế

Ví dụ :

PHP Code:

```
$string = 'This is a test of substr_replace ';\n$result = substr_replace($string, 'replaced', 3,4);\nvar_dump($result);\n//Xuất 'Thireplaced a test of substr replace '\n$result = substr_replace($string, 'replaced', 3, -4);\nvar_dump($result);\n//Xuất 'Thireplacedace '\n$result = substr_replace($string, 'replaced', 3,0);\n//Xuất 'Thireplaces is a test of substr_replace'\n$result = substr_replace($string, 'replaced',-3,5);\n\n//Xuất 'This is a test of substr replareplaced'
```

MỘT SỐ QUY ƯỚC: tham số trong hàm được tô màu đỏ tức là optional, tự chọn .
Mình viết các PART bằng WORD nên dấu ' ' thành ' ' << Khi copy thì script hok
hiểu đâu @@

STRING trong PHP

[PART2]

III/ Thao tác chuỗi khi chuẩn bị xử lý

a/ Thao tác chuỗi khi lấy từ form

Chuỗi lấy từ form cần phải được xử lý . Đôi khi 1 source code của chúng ta có thể run được, chạy layout xuất ra browser bình thường nhưng khi validate thì lại có rất nhiều lỗi , và nó ảnh hưởng nghiêm trọng tới SEO .

VD: thẻ
 mà viết thành
 <<<< 1 lỗi

 mà viết thành
 <<<< 1 lỗi khác .

Nhưng khi xuất ra browser thì nó đều xuống hàng

Còn ở form, giả sử người dùng sau khi insert 1 cuốn sách có tựa đề “ John ‘s life “. Sau đó khi bạn show tựa đề sách ra browser thì browser sẽ vẫn hiển thị là John ‘s life, khi click phải chọn view source bạn cũng sẽ thấy John ‘s life . Nhưng kí tự “ ‘ “ là 1 kí tự đặc biệt, nó phải được mã hóa ra “'” , để tránh bị lỗi kể trên. Ví thể chúng ta cần mã hóa dữ liệu chuỗi lấy từ form .

Để làm điều này chúng ta cần 1 số hàm như :

htmlspecialchars(): đổi 1 số lượng lớn các kí tự thành mã

htmlspecialchars(string, quotestyle): chỉ đổi 5 kí tự đặc biệt là nháy đôi (“), nháy đơn (‘) , dấu bé (<), dấu lớn (>) và dấu & .

QUOTESTYLE có các giá trị sau :

ENT_QUOTES: mã hóa cả 2 nháy ‘ và “

ENT_NOQUOTES: hok mã hóa nháy nào cả

ENT_COMPAT : default, chỉ mã hóa “

Chuyển đổi sang mã :

• “ => "

• ‘ => '

• & => &

• < => <

• > => >

b/Thao tác khi chuẩn bị đưa vào database:

Khi chuẩn bị đưa vào database, chúng ta cần phải tối ưu dữ liệu, càng ngắn càng tiết kiệm

VD :

PHP Code:

```
$conn = mysql_connect('localhost', 'root', '');
mysql_select_db('test', $conn);
$text = '                                This is to TEST
';
$sql = "insert into test(test) values ('$text')";
mysql_query($sql);
```

Vậy chúng ta đã insert 63 kí tự thay vì 15 chỉ kí tự . Thật là 1 sự phí phạm

Vậy để hạn chế điều này chúng ta dùng các hàm : **trim()**, **ltrim()**, **rtrim()** .

Trim(string, charlist): Hàm này có nhiệm vụ bỏ qua các khoảng trắng và 1 số kí tự khác từ 2 phía của string (hok có ở giữa string nha).

Một số kí tự nên nhớ trong charlist

• “\0” : NULL

• “\t” : tab ngang

• “\n” : xuống dòng

• “\r” : carriage return

• “ ” : khoảng trắng

• “\x0B” : tab đứng

Ltrim(string, charlist): chỉ bỏ qua bên trái

Rtrim(string, charlist): chỉ bỏ qua bên phải

Nếu không chỉ ra charlist thì mặc định sẽ bỏ qua cả 5 loại trên

1 vấn đề khác khi chuyển dữ liệu vào database có bao gồm các kí tự như (‘), (“), (\\), (NULL) thì database có thể nhầm lẫn với các kí tự điều khiển (control characters) . Control characters là những kí tự có nhiệm vụ như kết thúc, thay đổi, bắt đầu 1 câu lệnh hay 1 hàm nào đó . Vì thế để database phân tích những kí tự đó như là 1 kí tự thông thường , chúng ta cần phải thêm \ .

Ví dụ

PHP Code:

```
$conn = mysql_connect('localhost', 'root', '');
mysql_select_db('test', $conn);
$text = "          This is to TEST          ";
$sql = "insert into test(test) values ('$text')";
$result = mysql_query($sql);
if( $result){
    echo `Insert successfully`;
}
else{
    echo `Fail`;
}
//Code trên sẽ xuất "Fail" vì database hok hiểu " ` " .
```

Để cho database hiểu, chúng ta cần phải thêm “\” vào trước nó .

Thay “ ` “ thành “ \ ` “ ta sẽ được insert successfully .

Để làm điều này chúng ta có hàm :

Addslashes(string): thêm \ vào trước các kí tự kể trên .

Stripslashes(string): bỏ \ trước các kí tự kể trên.

PHP có thể tự động thêm \ cho chúng ta, nếu PHP đã làm vậy mà chúng ta còn thêm \ thì sẽ dư . Vậy chúng ta dùng hàm : get_magic_quotes_gpc() để test xem chế độ đó đã được bật chưa .

VD:

PHP Code:

```
if(get_magic_quotes_gpc()){
    echo 'On';
}
else{
    echo 'off';
}
```

Hoặc chạy script chỉ chứa :

PHP Code:

```
<?php
phpinfo();
?>
```

Rồi sau đó search từ khóa ‘ magic_quotes ‘ để xem off hay on ! Nếu nó on thì mỗi khi chúng ta thao tác xuất ra browser , phải dùng stripslashes() . Còn khi insert vào

database thì hok cần addslashes nữa .

Khi `magic_quotes_gpc()` off thì

Lưu ý : Cho dù đã addslashes hay chưa addslashes thì khi query từ database chúng ta cũng không phải stripslashes trước khi thao tác vì database tự động bỏ \ và lưu ' vào CSDL .

STRING trong PHP

[PART3]

IV/ So sánh chuỗi :

Để so sánh chuỗi trong PHP chúng ta có các hàm sau :

`strcmp(string1, string2)`: so sánh string 1 và string 2

Hàm này trả về :

- >0 : nếu string1 > string2
- <0 : nếu string1 < string2
- =0: string1 bằng string2

So sánh theo kiểu : '1' < '12' < '2' < '22' <<< quy luật trong tự nhiên

Ví dụ :

PHP Code:

```
$string1 = 'Hello World! ' ;
$string2 = 'Hello World! ' ;
var_dump(strcmp($string1, $string2));
//Xuất 0
```

Hàm identical với `strcmp()` nhưng hok phân biệt hoa-thường là `strcasecmp()`

Bên cạnh đó chúng ta có hàm :

`strnatcmp(string1, string2)`: Đây là hàm so sánh 2 chuỗi nhưng theo quy luật tự nhiên , ngoài ra, nó tương tự `strcmp()`

Tức là '1' < '2' < '12' < '22' .

Hàm identical với `strnatcmp()` mà hok phân biệt hoa-thường là : `strnatcasecmp()`

V/ Chuyển đổi chuỗi và một số hàm cắt, nối :

`strtoupper(string)`: convert chuỗi sang IN HOA

`strtolower(string)`: convert chuỗi sang viết thường .

`ucfirst(string)`: Viết in hoa kí tự đầu tiên của chuỗi

`ucwords(string)`: viết in hoa mỗi kí tự đầu tiên của từng chữ trong chuỗi

Lưu ý : Các hàm trên trả về chuỗi đã được chuyển đổi

VD:

PHP Code:

```
$string = ' This is some text ';//chuỗi đã được ucfirst sẵn rồi
$string_lower = strtolower($string);
// 'this is some text'
$string_upper = strtoupper($string);
// 'THIS IS SOME TEXT'
$string_first = ucwords($string);
//' This Is Some Text '
```

array explode('string delimiter', string ,limit): cắt chuỗi ra thành mảng.**VD:**

PHP Code:

```
$string = ' This is string that will be exploded ' ;
$array = explode(' ', $string);
var_dump($array);
/* Chúng ta sẽ có :
    array
    0 => string '' (length=0)
    1 => string 'This' (length=4)
    2 => string 'is' (length=2)
    3 => string 'string' (length=6)
    4 => string 'that' (length=4)
    5 => string 'will' (length=4)
    6 => string 'be' (length=2)
    7 => string 'exploded' (length=8)
    8 => string '' (length=0)
```

*/

Lưu ý : string delimiter hok thể là rỗng (") được**implode(separator,array):** nối chuỗi từ 1 mảng và các giá trị nối nhau bởi separator**VD:**

PHP Code:

```
$array = array('Hello', 'I', 'am', 'student');
$string = implode(' ', $array);
var_dump($string);
//Xuất 'Hello I am student'
```

Nếu separator ở đây rỗng thì sẽ rất khó nhìn bởi các chữ dính liền nhau

strtok(string, split): cũng là 1 hàm cắt chuỗi nhưng khác với explode, strtok cắt từng khúc một . Tức là chỉ cắt 1 lần mỗi khi gọi.**VD:**

PHP Code:

```
$string = ' This is some text that will be token ' ;
$part =strtok($string, ' ');
var_dump($part);
//Xuất ' This'
//Để lấy từng chữ :
while($part != false){
    echo "$part<br/>";
```

```
$part = strtok(' '); //Lần gọi thứ 2 hok cần phải nêu string ra nữa  
}
```

STRING trong PHP

[PART4]

VI/ Một số hàm tham khảo khác :

sha1(string), md5(string) : mã hóa chuỗi theo thuật toán tương ứng

nl2br(string): chuyển \n thành
 . Browser hok hiểu \n nhưng hiểu
 .

VD:

PHP Code:

```
$string = "This is first line \n This is second line <br/> ";  
//Nháy đôi vì phải hiểu \n đại diện cho cái gì chứ hok đơn giản là kí  
tự .  
echo $string;  
//Xuất 1 dòng duy nhất  
echo nl2br($string);  
//Xuất 2 dòng vì \n đã dc chuyển đổi
```

number_format(number, decimals, decimalpoint, separator):

Với :

- Decimals : số chữ số thập phân muốn xuất ra
- Decimalpoint: string phân cách giữa phần nguyên và phần thập phân
- Separator: string phân cách giữa phần trăm, phần nghìn, phần triệu,.....

VD:

PHP Code:

```
$number = 1000000;  
$number_formatted = number_format($number, 3, ',', '.');  
var_dump($number_formatted);  
// Xuất 1.000.000,000
```

str_pad(string, length, pad_string, pad_type):

Padtype:

STR_PAD_RIGHT

STR_PAD_LEFT

STR_PAD_BOTH

VD:

```
$str = 'Hello WORLD';  
$str_padded = str_pad($str, 20, '.');  
var_dump($str_padded);  
//Xuất 'Hello WORLD.....'
```

str_repeat(string, repeat): lặp lại string

VD:

PHP Code:

```
$string = 'Repeat ';  
var_dump(str_repeat($string, 5));  
//Xuất 'Repeat Repeat Repeat Repeat Repeat'
```

str_rot13(string): đẩy các kí tự trong string 13 đơn vị trong bảng chữ cái

VD:

PHP Code:

```
$string = 'abc';  
var_dump(str_rot13($string));  
//Xuất 'nop' <<<Viết bằng chữ cái ra sẽ thấy n-a = 13
```

str_shuffle(string): random các kí tự trong chuỗi string

VD:

PHP Code:

```
$string = 'abc';  
var_dump(str_shuffle($string));  
//Xuất 'bca' hoặc 'abc' hoặc 'acb' sau mỗi lần refresh
```

str_word_count(string, return, char): làm việc với số từ trong 1 chuỗi

Các giá trị có thể của **return**:

- 0: mặc định, trả về số từ trong chuỗi
- 1: trả về mảng tuần tự với các giá trị là các từ trong chuỗi
- 2: trả về mảng mà khóa là vị trí của mỗi từ trong chuỗi, giá trị của khóa đó là từ tại vị trí đó

Char: 1 số kí tự như & \$,....

Lưu ý: Hàm này hok cho phép các kí tự như ^ % # *... và những con số

VD 1:

PHP Code:

```
$string = "Word_1 word_2 & word_3 " ;  
var_dump(str_word_count($string, 0 ) ) ;  
// Trả về 3  
var_dump(str_word_count($string, 1));  
/* Trả về  
array  
0 => string 'Word' (length=4)  
1 => string 'word' (length=4)  
2 => string 'word' (length=4)  
*/  
var_dump(str_word_count($string, 1, '_'));  
/* Trả về  
array  
0 => string 'Word_' (length=5)  
1 => string 'word_' (length=5)  
2 => string 'word_' (length=5)  
*/
```

strrev(string): đảo chuỗi

VD:

PHP Code:

```
$string = ' ABC';  
var_dump(strrev($string));  
//Xuất 'BCA '
```

substr_count(string, substring, start, length): đếm 1 chuỗi con nào đó xuất hiện trong chuỗi gốc bao nhiêu lần

Start: vị trí xuất phát tìm kiếm

Length: tìm kiếm từ vị trí xuất phát, tìm trong 1 độ dài đơn vị

VD:

PHP Code:

```
$string = ' Hello WORLD, you are my world . ';  
var_dump(substr_count($string, 'world'));  
//Xuất 1 vì hàm này phân biệt hoa- thường  
$string1 = ' Hello world, you are my world. ';  
var_dump(substr_count($string1, 'world', 10));  
//Xuất 1 vì bắt đầu tìm từ vị trí chữ 'l'  
var_dump(substr_count($string1, 'world', 10, 1));  
//Xuất 0 vì bắt đầu tìm từ vị trí chữ 'l' và tìm đến hết chữ 'd' vì độ dài chỉ có 1
```

Wordwrap(string, width, break, cut): hàm giới hạn số kí tự trên 1 dòng . Trả về string đã bị break nếu thành công hoặc false nếu thất bại

Break: chỉ ra kí tự dùng để break. Mặc định là \n <<Chỉ view source mới thấy xuống dòng

Cut: chỉ định cắt chữ ra nếu dư (set TRUE) mặc định là FALSE .

VD:

PHP Code:

```
$string = ' This is some text. It's long text . ';  
echo wordwrap($string, 5);  
//Xuất bình thường nhưng khi click phải view source sẽ thấy mỗi dòng chỉ có 5 kí tự  
echo '<br/>';  
echo wordwrap($string, 5, '<br/>');  
//Xuất mỗi dòng 5 kí tự . Vì browser hiểu <br/> . Khi thay số 5 bằng 2 cũng cho kq tương tự vì cut = FALSE  
echo '<br/>';  
echo wordwrap($string, 2, '<br/>', TRUE);  
//Xuất mỗi dòng 2 kí tự, chữ bị cắt ra .
```

Mảng trong PHP [Part1]

I/ Mảng là gì ?

Trong PHP, biến là nơi lưu trữ dữ liệu . Nhưng 1 biến chỉ có thể lưu trữ 1 dạng dữ liệu và lưu trữ riêng lẻ .

Ví dụ:

PHP Code:

```
$a =3;  
$a =5;  
$a =10;
```

Ở đoạn code trên thì biến \$a cuối cùng sẽ có giá trị là 10 . Vậy chúng ta cần 1 phương thức lưu trữ khác thuận tiện hơn, dễ quản lí hơn . Và mảng ra đời để đáp ứng điều đó. Hình dung mảng là 1 cái tủ, trong cái tủ đó có nhiều ngăn kéo, mỗi ngăn kéo được đánh dấu bởi 1 con số hoặc 1 dòng chữ . Vậy để lấy món đồ trong 1 cái ngăn kéo , thì tất nhiên chúng ta cần 1 con số ứng với cái ngăn kéo ấy .

II/Khởi tạo mảng

Để khởi tạo mảng chúng ta dùng từ khóa **array**

Trong PHP có 3 cách khởi tạo mảng như sau :

- 1/ Khởi tạo 1 mảng rỗng, sau đó mới set giá trị
- 2/ Khởi tạo 1 mảng với 1 số giá trị ban đầu, sau đó thêm giá trị mới vào
- 3/ Khởi tạo mảng và set full giá trị cho mảng đó .

III/Phân loại mảng

Mảng được chia thành 3 loại chính :

- Mảng tuần tự
- Mảng không tuần tự
- Mảng đa chiều

1/ Mảng tuần tự

Mảng tuần tự là mảng mà các khóa của nó được đánh theo số . Vị trí đầu tiên của mảng luôn là 0

Ví dụ : `$array_name = array('a', 'b', 'c', 'd');`

Ở trên, để lấy giá trị 'a' , ta dùng cú pháp : `$array_name[0];`

Tương tự, để lấy giá trị 'b', 'c', 'd' : `$array_name[1], $array_name[2], $array_name[3]`

a/ Khởi tạo mảng tuần tự:

i/ Tạo mảng rỗng rồi thêm giá trị

```
$array = array();
```

```
$array[] = 'a';
```

```
$array[] = 'b' ;
```

ii/ Tạo mảng sau đó thêm giá trị mới vào

```
$array = array('a');
```

```
$array[] = array('b');
```

iii/ Tạo mảng full giá trị

```
$array = array('a','b');
```

Lưu ý: Nếu muốn khởi đầu mảng hok phải là 0 mà là 1 chúng ta làm như sau :

```
$array = array('1'=>'a', 'b', 'c' );
```

b/ Duyệt mảng tuần tự

Chúng ta sử dụng 1 số hàm sau để duyệt mảng tuần tự :

foreach(), for()

FOR: Giả sử chúng ta biết mảng \$array_name có \$count phần tử

```
for($i = 0; $i < $count ; $i++){
```

```
//viết code thao tác với phần tử mảng ở đây
```

```
}
```

Giải thích:

Hàm for dùng để duyệt từng phần tử của mảng, để dùng cách này chúng ta phải biết mảng này có bao nhiêu phần tử . For sẽ có nhiệm vụ lặp qua từng phần tử một, bắt đầu từ vị trí số 0 cho đến \$count - 1.

Ví dụ :

PHP Code:

```
for($i = 0; $i < $count; $i++){  
    echo "Giá trị của phần tử $i là :  $array_name[$i] <  
br/>";  
}
```

```
// Code xuất từng giá trị của mảng
```

FOREACH

```
foreach($array_name as $value){
```

```
// Viết code thao tác với phần tử mảng ở đây
```

```
}
```

Giải thích :

Hàm foreach cũng duyệt từng phần tử của mảng, với mỗi phần tử nó đi qua, nó sẽ gán giá trị của phần tử đó cho biến \$value .

Ví dụ :

PHP Code:

```
$i = 0;  
foreach($array_name as $value){  
    $i++; //Tăng lên 1 mỗi khi foreach duyệt qua 1 phần  
tử  
    echo "Giá trị của phần tử $i là :  $value <br/>";  
}
```

```
// Code xuất từng giá trị của mảng
```

2/ Mảng không tuần tự

Mảng không tuần tự là mảng mà các khóa của nó được đánh theo chữ . Hình dung 1 ngăn tủ, mà các ngăn kéo của nó không phải là con số nữa mà là những từ ngữ có nghĩa .

Ví dụ : ngăn kéo tên 'Giày' chứa giày , ngăn kéo tên 'quần áo' chứa quần áo.

=> Rõ ràng để truy cập những vật ở bên trong các ngăn kéo này sẽ dễ hơn các ngăn kéo đánh số thứ tự !

a/ Khởi tạo mảng không tuần tự :

- \$array_ktt = array('a'=> 'a_123','b'=>'b_123');

b/ Duyệt mảng không tuần tự :

Chúng ta dùng các hàm như : list(), each(), while(), foreach() để duyệt mảng không tuần tự

FOREACH():

PHP Code:

```
        foreach($array_ktt as $value){
            echo " $value | ";
        }
//Code trên dùng để thao tác chỉ với các GIÁ TRỊ của mảng không tuần tự

        foreach($array_ktt as $key=>$value){
            echo " $key => $value | ";
        }
//Code trên dùng để thao tác với cả GIÁ TRỊ và KHÓA của mảng không tuần tự
```

WHILE :

```
while(list($key,$value) as each($array_ktt)){
//Code ở đây
}
```

Code trên có nghĩa: với mỗi giá trị mà while duyệt qua, khóa và giá trị của \$array_ktt lần lượt được gán vào \$key và \$value .

```
while($element as each($array_ktt)){
//Code ở đây
}
```

Code trên có nghĩa: với mỗi giá trị mà while duyệt qua, phần tử của \$array_ktt lần lượt được gán \$element . \$element ở đây đóng vai trò như là 1 cái mảng không tuần tự để lưu \$key và \$value của mỗi \$array_ktt mà while duyệt qua .

Ví dụ :

PHP Code:

```
while($element as each($array_ktt)){
    echo $element['key'];
    echo ' - ';
    echo $element['value'];
    echo '<br/>';
}
//Code trên có nghĩa: với mỗi giá trị mà while duyệt qua, phần tử của $array_ktt lần lượt được gán $element . $element ở đây đóng vai trò như là 1 cái mảng không tuần tự để lưu khóa và giá trị của mỗi $array_ktt mà while duyệt qua .
```

3/ Mảng đa chiều

Hình dung 1 cái tủ to có các ngăn kéo, ở trong 1 số ngăn kéo lại chia thành 2-3 cái ngăn kéo nhỏ khác . Hay cũng có thể nói là : trong ngăn kéo có 1 chiếc tủ nhỏ khác

Ví dụ:

```
$multi_array_tt = array('a', 'b', array('c','d'),'e');
$multi_array_ktt = array('a'=>'a_123', 'b'=>'b_123',
'small_array'=>array('c'=>'c_123','d'=>'d_123'),'e'=>'e_123');
```

Vậy để truy cập vào ngăn kéo cái tủ nhỏ ấy, ta phải: Truy cập vào ngăn kéo của tủ lớn chứa tủ nhỏ, sau đó truy cập vào ngăn kéo của tủ nhỏ .

```
echo " $multi_array_tt[2][0] "; //Xuất ra 'c'
// Truy cập vào mảng là phần tử thứ 3 của mảng $multi_array_tt, sau đó lấy phần tử thứ 1 của mảng ấy ra .
```

```
echo " $multi_array_ktt['small_array']['c'] "; //Xuất ra 'c_123'
```

// Truy cập vào mảng là phần tử có khóa 'small_array' của mảng \$multi_array_ktt, sau đó lấy phần tử có khóa 'c' của mảng ấy ra .

Duyệt mảng đa chiều: (chỉ dùng cho mảng tuần tự)

PHP Code:

```
$multi_array_test= array( array('a', 'b', 'c'),  
                           array('a1', 'b1',  
'c1'),  
                           array('a2', 'b2'  
, 'c2') );  
// Bạn hình dung mảng $multi_array_test như trên là 1 table có 3 cột và  
3 dòng  
for($row = 0; $row <3; $row++){  
    for($column =0; $column < 3; $column++){  
        echo '|'.$multi_array_test[$row][$column];  
    }  
    echo '<br/>';  
}
```

IV/ Con trỏ mảng :

Mảng được thao tác dựa trên vị trí con trỏ của mảng, khi mảng vừa được tạo ra, con trỏ nằm ở vị trí đầu tiên . Khi thêm giá trị mới vào mảng thì con trỏ vẫn hok di chuyển . Để di chuyển hay thao tác con trỏ chúng ta dùng 1 số hàm như prev(), next(), end(), reset(), each(), current() . Chúng ta sẽ tìm hiểu thêm ở phần II .

Mảng trong PHP [Part2]

IV/ Một số hàm cần nhớ trong mảng:

```
$array1 = array('b','a','c');  
$array2 = array('1','3','6');  
$array3 = array('b'=>'d_123', 'a'=>'f_123' , 'c'=>'e_123');
```

Các hàm thông dụng nhất :

sort(\$name_of_array, SORT_...): sắp xếp lại các phần tử trong mảng (theo bảng chữ cái hoặc theo số). Luật : 1<11<12<2<21.

Các giá trị có thể của tham biến thứ 2:

SORT_REGULAR

SORT_STRING

SORT_NUMERIC

VD:

```
sort($array1)
```

=>\$array1 bây giờ là array('a','b','c')

Lưu ý : hàm này sắp xếp theo thứ tự A<Z<a<z .

Sort theo quy luật tự nhiên : tức là 1<2<3<20....

natsort(\$array) : phân biệt chữ hoa và chữ thường

natcasesort(\$array) : hok phân biệt chữ hoa và chữ thường

VD:

PHP Code:

```
$array = array('a1','a2','a11','a22','A24');
sort($array);
var_dump($array);
natsort($array);
var_dump($array);
natcasesort($array);
var_dump($array);
/* Nhận được kết quả sau :
array
  0 => string 'A24' (length=3)
  1 => string 'a1' (length=2)
  2 => string 'a11' (length=3)
  3 => string 'a2' (length=2)
  4 => string 'a22' (length=3)
array
  0 => string 'A24' (length=3)
  1 => string 'a1' (length=2)
  3 => string 'a2' (length=2)
  2 => string 'a11' (length=3)
  4 => string 'a22' (length=3)
array
  1 => string 'a1' (length=2)
  3 => string 'a2' (length=2)
  2 => string 'a11' (length=3)
  4 => string 'a22' (length=3)
  0 => string 'A24' (length=3)
*/
```

ksort(\$array): sắp xếp theo khóa của các phần tử trong mảng tăng dần

VD:

ksort(\$array3)

=>\$array3 bây giờ là array('a'=>'f_123', 'b'=>'d_123', 'c'=>'e_123')

Asort(\$array): sắp xếp theo giá trị của các phần tử trong mảng tang dần

VD:

asort(\$array3)

=>\$array3 bây giờ là array('b'=>'d_123', 'a'=>'f_123', 'c'=>'e_123')

Đổi lập với **ksort(\$array)** là **krsort()** : sắp xếp theo khóa với các phần tử trong mảng giảm dần

asort(\$array) là **arsort()** : sắp xếp theo giá trị với các phần tử trong mảng giảm dần

shuffle(\$array): thay đổi vị trí các phần tử mảng 1 cách random

VD:

shuffle(\$array1)

=> \$array1 bây giờ có thể là array('a', 'b', 'c') hay array('c', 'b', 'a') , v.v

Array_reverse(\$array1) : trả về mảng đảo ngược

VD:

\$array1_reverse = array_reserver(\$array1)

=>\$array1_reverse giờ là array('c','a','b')

Như đã nói ở phần 1, nhiều khi chúng ta phải thay đổi vị trí con trỏ mảng để thao tác theo ý muốn. Để làm điều đó ta dùng các hàm:

Prev(): lùi con trỏ 1 phần tử

next(): tiến con trỏ 1 phần tử,

reset(): reset con trỏ về vị trí 0

end(): đưa con trỏ đến vị trí cuối

Lưu ý: Các hàm trên đều trả về giá trị của nơi mà con trỏ đang đứng

current(): xuất giá trị của vị trí hiện tại của con trỏ

each(): trả về key và giá trị của vị trí con trỏ đang đứng, sau đó tiến con trỏ lên 1 phần tử

Hàm đếm phần tử mảng:

Count(\$array)

Array_count_values(\$array): Hàm này đếm số lần xuất hiện của 1 giá trị trong mảng.

Trả về kiểu dữ liệu mảng

Vd:

PHP Code:

```
<?php
$array = array('a','b','c','d','a','c');
echo current($array); // Xuất 'a' do khi mới tạo, con trỏ nằm ở vị
trí đầu
echo next($array); // Xuất 'b'
echo '<br/>';

$value = end($array); //Chuyển con trỏ về cuối( trước 'c')
while($value){
    echo " $value ";
    $value = prev ($array); //Sau mỗi lần lặp, lùi con trỏ 1 phần t
    }
echo '<br/>';
echo reset($array); //Đưa con trỏ về vị trí đầu

$each = each($array); //Lấy giá trị và khóa của vị trí đầu, trả về
kiểu dữ liệu mảng
var_dump($each);

var_dump(array_count_values($array)); // Trả về kiểu dữ liệu mảng .
?>
```

Và dưới đây là 2 hàm khá quan trọng trong thao tác mảng

Explode ('string delimiter', 'string input', limit): Cắt chuỗi dựa vào kí tự chỉ định trước, limit giới hạn số phần tử của mảng trả về

Extract(\$array, extract_type, prefix): Tạo biến với tên là khóa của mảng và biến đó có giá trị là giá trị của khóa đó. Prefix có thể dịch là tiền tố

Có 7 loại extract:

- **EXTR_OVERWRITE**: mặc định. Nếu khóa của mảng trùng với tên biến trong script thì ghi đè

- **EXTR_SKIP** : Nếu khóa của mảng trùng với tên biến trong script thì bỏ qua khóa đó .
- **EXTR_PREFIX_SAME** : nếu khóa của mảng trùng với 1 khóa khác thì thêm tiền tố vào khóa đó . Với loại này phải chỉ ra prefix ở tham số thứ 3
- **EXTR_PREFIX_ALL**: Thêm tiền tố vào mọi khóa. Với loại này phải chỉ ra prefix ở tham số thứ 3
- **EXTR_PREFIX_IF_EXISTS**: Nếu khóa của mảng trùng với tên biến của script thì thêm tiền tố vào khóa đó. Với loại này phải chỉ ra prefix ở tham số thứ 3
- **EXTR_PREFIX_INVALID**: Nếu khóa của mảng vi phạm qui tắc đặt biến thì thêm tiền tố vào khóa đó . Với loại này phải chỉ ra prefix ở tham số thứ 3
- **EXTR_REFS**: biến được tạo ra sẽ tham chiếu đến khóa của mảng , mọi thay đổi trên biến đó sẽ ảnh hưởng đến giá trị mà khóa đó đại diện trong mảng

Ví dụ :

PHP Code:

```
$array = array('a'=>'a_123', 'a'=>'b_123', 'c'=>'c_123' , '1'=>
'INVALID');
$array_1= array('a'=>'a_123', 'd'=>'d_123', 'e'=>'e_123');
$a = 4;
(1)extract($array,EXTR_PREFIX_SAME,'abc');
//Tạo biến $a = 'a_123' và $abc_a = 'b_123'
var_dump($a);
var_dump($abc_a);
(2)extract($array,EXTR_PREFIX_ALL,'abc');
//Tạo biến $abc_a = 'b_123' (đã bị ghi đè )và $abc_c = 'c_123'
var_dump($abc_a);
var_dump($abc_c);
(3)extract($array_1,EXTR_PREFIX_IF_EXISTS,'ade');
//Tạo biến $ade_a = 'a_123' và $d = 'd_123' và $e= 'e_123'
var_dump($ade_a);
var_dump($d);
var_dump($e);
(4)extract($array,EXTR_PREFIX_INVALID,'abc');
//Tạo biến $abc_12 = 'INVALID' và $a= 'b_123' ( đã bị ghi đè) , $c = '
c_123'
var_dump($a);
var_dump($abc_12);
var_dump($c);
(5) extract($array,EXTR_REFS);
//Tạo biến $a = 'b_123' và $c = 'c_123'
var_dump($c) ; //Xuất c_123
$c =5;
var_dump($array['c']) // Xuất ra 5 => Thay đổi giá trị trong
mảng
```

Khi thử code thì làm 1 trong 5 code trên thôi nhé !

```
Cho 1 mảng : $array_test_tt = array('a', 'b', 'c');  
$array_test_ktt = array('a'=>'a_123', 'b'=>'b_123', 'c'=>'c_123');
```

bool array_key_exists('key_to_search', \$name_of_array); : tìm 1 khóa có trong mảng hay không

```
//Bool là kiểu data trả về true hay false  
VD: array_key_exists('0', $array_test_tt); Trả về TRUE  
array_key_exists('a', $array_test_ktt); Trả về TRUE  
array_key_exists('d', $array_test_ktt); Trả về FALSE  
Để tìm 1 giá trị trong mảng :  
Bool in_array($array, 'value');  
VD: in_array($array_test_tt, 'd') //trả về FALSE  
in_array($array_test_tt, 'a') // Trả về TRUE
```

mixed array_search('value_to_search', \$name_of_array);

```
// Mixed là hok phải là 1 kiểu dữ liệu, ý mình là nó có thể trả về string khi là mảng hok  
tuần tự và trả về số khi là mảng tuần tự  
VD: array_search('a', $array_test_tt); //Trả về 0  
array_search('a_123', $array_test_ktt); Trả về 'a'
```

string array_pop(\$name_of_array);

```
// Hàm này lấy dữ liệu cuối cùng ra khỏi mảng  
VD: array_pop($array_test_tt); //Trả về 'c' và mảng $array_test_tt chỉ còn lại 'a' và 'b'  
array_pop($array_test_ktt); //Trả về 'c_123' và mảng $array_test_ktt chỉ còn lại 'a123' và  
'b123'
```

Các hàm dưới đây chỉ nên dùng cho mảng tuần tự

int array_push(\$name_of_array, 'value_to_push');

```
// Hàm này thêm 1 phần tử vào cuối mảng và trả về tổng số phần tử của mảng sau khi  
thêm  
VD: array_push($array_test_tt, 'd'); // Trả về số 4 và mảng $array_test_tt lúc này có thêm  
phần tử thứ 3 là 'd'
```

string array_shift(\$name_of_array)

```
// Hàm này lấy 1 phần tử Ở ĐẦU mảng ra khỏi mảng  
VD: array_shift($array_test_tt); //Trả về chữ 'a' . Và khóa của mảng lần lượt là 0 ứng với  
'b' , 1 ứng với 'c'
```

int array_unshift(\$name_of_array, 'value_to_unshift')

// Hàm này thêm 1 phần tử vào ĐẦU mảng và trả về tổng phần tử trong mảng
VD: array_unshift(\$array_test_tt,'d'); //Trả về số 4 . Và khóa của mảng lần lượt là 0 ứng với 'd' , 1 ứng với 'a', 2 ứng với 'b', và 3 ứng với 'c'

int array_rand(\$name_of_array)
// Hàm này random trả về 1 khóa trong mảng .
VD: array_rand(\$array_test_tt) // Có thể trả về 0 hay 1 hay 2

Hướng dẫn code trang news cơ bản

Lúc trước, khi mình tự học PHP, chưa đăng kí ở Việt chuyên, đọc sách, lên mạng xem tutorial, dành tg hầu như cả ngày cho PHP nhưng những kiến thức mình có được đều rời rạc . Như mảng, string, file, kết nối database , v.v . Vậy để xây dựng 1 trang web đơn giản thì làm thế nào ? Cần những kiến thức gì ? Database phải ra làm sao ? Quy trình như thế nào ? Và mình không hình dung được gì cả, hok trả lời được gì cả lên mạng download project về thì như sấm đánh bên tai, các dự án đó viết bằng AJAX và JQUERY, CSS v.v khiến bản thân những ng mới học rất dễ bị hoang mang, và mất tự tin . Thực ra mình chỉ muốn làm 1 trang web đơn giản , dùng PHP và MySQL thêm tí CSS để hiểu được mình đã học được gì và muốn áp dụng kiến thức học được vào bài tập cụ thể . Bởi vậy, lúc coi project của các bạn khác nhiều lúc mình muốn bỏ PHP luôn, cho đến khi mình biết được trung tâm Việt Chuyên , và bây giờ thì rất an tâm với sự dẫn dắt của thầy Huy thì sức trâu của chúng ta bỏ ra mới xứng đáng @@ .

Vậy để giúp những bạn tự học PHP, có lẽ cũng đang search google tìm cách thức làm 1 trang news đơn giản như mình đã từng làm , mình viết bài hướng dẫn sơ bộ này .

Code trang tin tức cơ bản

A.1 số tính năng cơ bản của trang news :

Trang admin : đăng nhập, đăng xuất, cho phép thêm category, xóa category, thêm tin tức, xóa tin tức , list categories và list tin tức , thêm người dùng, xóa người dùng, list người dùng . Bên cạnh đó cho phép sửa category, sửa news đã post , sửa level của người dùng .

Vậy folder admin có các file:

Index.php <<< layout trang admin

Login.php

Logout.php

add_cate.php,

add_news.php

add_user.php

list_cate.php

list_news.php

list_users.php

edit_cate.php

edit_news.php

edit_user.php

delete_cate.php

delete_news.php

delete_user.php

Trang layout : index.php ở ngoài ngang cấp với folder admin (khác với index.php ở trang admin) để dữ liệu từ database ra browser . Trang này tùy mỗi ngườivà là công đoạn đơn giản trong ứng dụng .

B. Hướng dẫn:

Kiến thức cần biết khi viết ứng dụng này

Form HTML, PHP cơ bản(mảng, string, biến, mysql, thao tác với mysql bằng PHP), PHP SESSION, PHP UPLOAD , SQL cơ bản: select, delete, insert, update . Hiểu cách lấy dữ liệu qua GET và POST . Biết cách tạo layout của 1 trang web đơn giản bằng require_once header, footer, body ,... Và nếu có thêm 1 tí CSS, hiểu code phân trang thì tốt

Database thiết kế như thế nào ?

Vì là ứng dụng đơn giản nên database cũng đơn giản :

Database **news** , có các table là :

User(**user_id**, user_name, user_password,user_level)

News(**news_id**, news_category, news_title,news_author, news_info, news_full, news_check, news_image, **news uid**, **news cid**)

Categories(**category_id**, category_name)

Ở trên gọi là 1 schema của database, nó tóm tắt các table và các cột có trong database, và mình viết chữ đỏ là khóa chính, gạch dưới là khóa ngoài .

Tại sao ở trên có news_author, news_category mà còn cần news_id, news_cid ? Vì news_author và news_category là những field có thể thay đổi được, như hnay bạn viết tin với ký danh là 'ABC' ngày mai bạn lấy ký danh khác là 'BCD' vậy để thống kê số tin bạn viết làm bằng cách nào ? news_id giúp chúng ta giải quyết điều đó vì dụ bạn lấy ký danh nào thì news_id chỉ có 1 mà thôi . Tương tự với news_cid

Giải thích: user có nhiều loại, chúng ta phải có trường user_level để xác định xem đó là admin, hay là user bình thường ? Nếu chỉ có 2 level thì set level admin là 2, user bt là 1 . Và chỉ có level bằng 2 thì mới cho truy cập vào vùng admin .

Cột news_check trong table news là để check xem news đó có hợp lệ không, đâu phải tin nào cũng được post lên ?

Cột news_image thường là optional tùy theo ý muốn người dùng có muốn dùng ảnh đại diện hay không

Quy trình :

Nếu là user bình thường thì chỉ được xem trang chủ, nếu là admin thì cho phép vào trang giao diện admin, tại đây có 1 menu tới các trang add, list, v.v . Tại các trang list, cho phép admin sửa, xóa . Nếu không phải admin mà đăng nhập thì đá về lại trang đăng nhập .

Một số điều cần lưu ý khi viết code :

Index.php: Trang admin này chỉ cần show ra các đường link để tới chỗ để xóa, sửa, v.v . Hok cần cầu kì vì chỉ có admin mới thấy được, trang layout cho user xem thì mới cần CSS phức tạp để cho đẹp . Tùy bạn layout trang index.php này, có thể là 1 table chẳng hạn .

Các trang xử lí form (add_) :

- **Nhớ là kiểm tra người dùng có nhập đầy đủ thông tin hay không** . Lưu ý là trong PHP chúng ta phải tạo biến trước khi dùng . Ví dụ, chỉ khi tồn tại biến \$user và \$pass thì mới tiến hành kiểm tra trong database và nếu có thì cho đăng nhập.

Tức là code:

PHP Code:

```
if($_POST['user'] !=NULL){
    $user = $_POST['user'];
}
else
    echo 'Ban chua nhap ten ';
    if($_POST['password'] !=NULL){
        $pass = $_POST['password'];
    }
    else
        echo 'Ban chua nhap password ';
if($user && $pass ) { }
```

Nhưng nếu người dùng bỏ trống 1 trong 2 field trên rồi bấm submit thì \$_POST['user'] và \$_POST['pass'] là NULL và chúng ta không tiến hành gán biến \$user và \$pass <<<

tức là 2 biến này chưa tồn tại . Vì thế dòng if() ngay sau đó sẽ bị lỗi . Vì thế chúng ta cần phải set giá trị NULL ban đầu cho 2 biến đó :

PHP Code:

```
$user=$pass='';
if($_POST['user'] !=NULL){
    $user = $_POST['user'];
}
else
    echo 'Ban chua nhap ten ';
    if($_POST['password'] !=NULL){
        $pass = $_POST['password'];
    }
else
    echo 'Ban chua nhap password ';
if($user && $pass ) { }
```

- **Các bạn nên viết form và cách xử lý form trên cùng 1 trang.** Ví dụ form add_cate tại add_cate.php sẽ có action cũng là add_cate.php . Ở phần trên của form bạn kiểm tra xem người dùng đã nhấn submit nộp thông tin chưa, hay chỉ là đang cần điền vào form ?

Ta dùng code :

PHP Code:

```
if(isset($_POST['ok'])) {
    // 'ok' là tên trường submit <input type = 'submit' value = 'login'
    name = 'ok' />
    //Xử lý form ở đây
}
```

- **Tại trang login:** nếu người dùng đã login được và chúng ta đã xác nhận được đó là admin, thì cấp session cho admin đó

Cu thể là :

PHP Code:

```
$_SESSION['username'] = $username ( biến $username này lấy từ form )
$_SESSION['level'] = $level
$_SESSION['u_id'] = $user_id
//Cả 2 $level và $user_id chúng ta phải lấy từ database về .
//Sở dĩ tạo 3 cái $_SESSION như thế vì sau này trong script khác chúng ta cần biết là ai đang truy cập, u_id là gì ?
```

- **Tại tất cả các file trong trang admin trừ login** , bạn phải kiểm tra xem người dùng có phải là admin không ? Nếu phải thì cho truy cập, không phải thì đá về trang login.php .

Vậy chúng ta có 1 code dùng rất nhiều trong phần admin là:

PHP Code:

```
if($_SESSION['level'] != '2'){
    header('location: login.php');
```

```

        exit();
    }

```

1 code khác cũng dùng khá nhiều là : connect database

PHP Code:

```

$conn = mysql_connect('localhost', 'root', '');
mysql_select_db('news', $conn);

```

Vậy 2 code trên đem bỏ vào 2 file, bỏ 2 file vào folder 'includes' rồi sau này dùng 'require_once()' để dùng lại .

- **1 số lưu ý với các file edit:**

Đồ dữ liệu đã có sẵn từ database ra form rồi cho phép người dùng sửa đổi . Ví dụ : tên news lúc post lên là 'NEWS'. Thì khi nhấn vào edit phải xuất ra chữ NEWS ở news_title chứ hok được bỏ trống .

Với file edit và add , chúng ta có thể dùng bộ soạn thảo để người dùng có thể soạn thảo tin tức như trong word . Chúng ta download file ckeditor.rar ở bên dưới, giải nén ra rồi bỏ thư mục ckeditor vào ngang cấp với các file edit_ , add_ .
Sau đó , chúng ta include ckeditor vào bằng cách :

PHP Code:

```

<script type="text/javascript" src="ckeditor/ckeditor.js"></script>
//Đặt ở trên <form > nhé

```

Nếu muốn phần nào có bộ soạn thảo thì chèn vào sau phần đó

PHP Code:

```

<script type="text/javascript">
    CKEDITOR.replace( 'name_of_field' );
</script>

```

VD:

....

PHP Code:

```

<tr>
    <td>Full</td>
    <td><textarea name='txtfull' cols='40' rows='15'><?php echo $data2['news_full'];?></textarea></td>
</tr>
<script type="text/javascript">
    CKEDITOR.replace( 'txtfull' );
</script>

```

....

=> Full sẽ có bộ soạn thảo cho người dùng

DOWNLOAD CKEDITOR: <http://www.mediafire.com/?ah85w2j03ddjdm6>

HƯỚNG DẪN CODE TRANG BÁN SÁCH

Tiếp theo mình xin hướng dẫn các bạn hình dung cách code 1 trang bán sách .

A. Mô tả ứng dụng bán sách

Nếu là khách (hok đăng nhập) thì sẽ được :

Chọn chuyên mục sách, khi click vào chuyên mục thì liệt kê các sách có trong chuyên mục . Khi click vào sách nào thì liệt kê thông tin của sách gồm tác giả , ISBN, ngày xuất bản, giá tiền , v.v . Tại đây cho phép người dùng thêm sách vào giỏ hàng . Sau khi thêm vào giỏ hàng có thể tạo nhiều option cho người dùng lựa chọn , hoặc là thanh toán -> cho người dùng nhập thông tin để ship sách , hoặc là tiếp tục mua sách .

Bước thanh toán có 2 giai đoạn :

1/ Nhập thông tin khách hàng, địa chỉ thanh toán, địa chỉ ship sách, v.v . Ứng dụng nhỏ, chúng ta để địa chỉ khách hàng là nơi ship sách luôn cho tiện

2/ Chọn phương thức thanh toán, điền mã thẻ (ví như thẻ CREDIT CARD, VISA CARD, v.v..)

Vậy, nếu người dùng click thanh toán, và sau đó hok thanh toán tiếp bước 2 mà chọn tiếp tục mua sách -> thêm vào giỏ hàng-> thanh toán <<<< Không yêu cầu người dùng nhập thông tin giai đoạn 1 nữa

Tại trang hiển thị giỏ hàng : cho phép người dùng thay đổi số lượng sách, và click nút ‘ Save ‘ để cập nhật, nếu người dùng nhập 0 thì sách đó sẽ được bỏ .

1 số lưu ý :

Khi người dùng đang xem list sách trong chuyên mục A thì nên có đường link dẫn khách qua trang list chuyên mục để ng dùng xem list sách của chuyên mục khác .

Tại trang xuất thông tin sách cụ thể, thì nên có 1 đường link dẫn khách qua trang list sách của chuyên mục chứa sách hiện tại .

Gợi ý : Tại 2 trang trên đều có 1 nút : ‘ Continue Shopping ‘ nhưng chức năng khác nhau .

Nếu đăng nhập dưới quyền user : chỉ khác khách ở chỗ bước thanh toán hok có giai đoạn 1 .

// Có thể bỏ hẳn user, chỉ admin và khách thôi cũng được

Nếu đăng nhập dưới quyền admin :

Dẫn admin tới trang menu list 1 số quyền của admin là :

Đến trang chính mua bán sách(<< mục đích sửa và xóa sách hay chuyên mục có sẵn)

Thêm chuyên mục

Thêm sách

List các users và level

Tại mỗi trang phải cho phép admin log out

Tại các trang list chuyên mục hay sách cho phép admin sửa, xóa, thêm, v.v

B. HƯỚNG DẪN

Kiến thức cần biết :

Để làm được ứng dụng này mình nghĩ các bạn cần chuẩn bị :

- PHP cơ bản, hiểu rõ mảng, hiểu rõ GET POST trong form HTML
- PHP SESSION
- Database MySQL select, delete, insert, update, v.v cơ bản
- SQL cơ bản
- Biết cách layout ghép header, footer, body

Quy trình :

Chúng ta cần các file sau :

(0) login.php : đăng nhập

(1)Index.php : list các chuyên mục

(2)List_books.php : list các sách trong chuyên mục

(3)Book_info.php : xuất thông tin sách

(4) Show_cart.php: hiển thị giỏ hàng

(5) check_out.php : hiển thị form để người dùng nhập thông tin .

(6) purchase.php : xử lý dữ liệu nhận từ người dùng (từ giai đoạn 1)

(7) process.php: xử lý dữ liệu nhận từ người dùng (từ giai đoạn 2)

(8) logout.php : đăng xuất

Folder admin :

Ad(0) admin.php : list các quyền của admin

Ad(1) delete_cate.php : xóa chuyên mục

Ad(2) delete_book.php: xóa sách

Ad(3) edit_cate.php : sửa chuyên mục

Ad(4) edit_book.php: sửa sách

Ad(5) insert_cate.php: thêm chuyên mục

Ad(6) insert_book.php : thêm sách

Ad(7) list_users.php : list các user và level

Ad(8) edit_user.php : chỉnh sửa

Ad(9) delete_user.php : xóa user

Nếu là khách (hok log in mà vào thẳng (1)) :

(1) ->(2) -> (3)

Nếu tại bước (3) , khách ưng ý, thêm vào giỏ : (3) -> (4)

Nếu tại bước (3), khách hok muốn mua, click 'continue' : (3) ->(2)

Nếu là user : dẫn user đi như khách nhưng bỏ thanh toán giai đoạn 1 .

Nếu là admin : (đã log in ở (0))

(0) -> Ad(0) [Ad(5), Ad(6), (1) , Ad(7)] .

[] <<< ý nghĩa là Ad(0) có thể dẫn tới các trang ở trong .

Lưu ý : Nếu admin chọn (1) thì đi y như khách, nhưng tại mỗi trang phải có đường dẫn để logout hay sửa, xóa chuyên mục /sách . Các trường hợp khác có lẽ các bạn cũng có thể hiểu cách dẫn admin đi như thế nào !!

Một số điều lưu ý khi dẫn đường cho admin:

- Tại các trang liên quan đến admin, luôn có sự kiểm tra xem người đang truy cập có phải admin hay không và luôn có link dẫn admin về lại Ad(0)
- Tại các trang list luôn có link dẫn admin đi sửa/ xóa
VD: Admin đang coi sách A, thì có link đi sửa/xóa sách A.

Database thiết kế như thế nào ?

Tạo DATABASE books với các table sau:

USER(user_id, username, password, level)

BOOKS(isbn, author, title, cate_id, price, description)

CATEGORIES (cate_id, cate_name)

CUSTOMERS(cus_id, cus_name, cus_address, cus_city, cus_state, cus_zip, cus_country)

ORDERS(order_id, cus_id, amount , date , confirm)

ORDER_ITEMS (order_id, order_items, item_price, quantity)

Giải thích :

- order_id ở ORDER_ITEMS không phải là khóa chính vì giả sử như bạn mua 5 cuốn sách giống nhau, tiến tới thanh toán giai đoạn 1, thì hok lẽ mỗi dòng trong database sẽ lưu 1 cuốn sách ? (1,2,3,4,5)
Thay vào đó, order_id ở đây sẽ là 1 số (1) , quantity là 5 .
Hoặc nếu bạn mua 3 cuốn A 2 cuốn B , thì order_id sẽ là (X ở đây là id , 1 con số)
:
X, Sách A, 1\$, 3
X, Sách B, 2\$, 2
Vậy sẽ dễ làm việc hơn với bảng ORDERS vì lúc này chỉ cần 1 dòng để lưu id X
- Level có thể có 2 cấp . 1 là user , 2 là admin .

Một số điều quan trọng :

1/ Code kiểm tra có phải là admin hay không và code kết nối database được sử dụng rất nhiều nên để vào 2 file bỏ vào thư mục includes để sau này dùng require_once() xài lại .

2/ Nếu là user => ẩn các tính năng như xóa, sửa ,v.v . Vì thế trong mỗi script (1)->(3) chúng ta phải kiểm tra, nếu đó là admin thì hiện, hok là admin thì ẩn .

3/ Nếu là admin thì đừng cho admin mua hàng ^^ .

4/ Khi xác nhận 1 ng là user (có thể là admin), cấp cho ng đó 1 phiên làm việc gồm username, level và u_id .

1 số gợi ý :

Hiện thị giỏ hàng : giả sử ng mua add sách A vào giỏ, kiểm tra xem có tồn tại sách A trong giỏ hok, nếu có thì tăng lên 1 , nếu hok thì thêm sách A vào .

Code :

PHP Code:

```
if(!isset($_SESSION['cart'])){
    $_SESSION['cart'] = array();
// Nếu người dùng chưa mua hàng, tạo 1 session là 1 mảng
// Mảng này ý đồ chỉ chứa (isbn) => (quantity)
}
if(isset($_SESSION['cart'][$isbn])){
    $_SESSION['cart'][$isbn] ++;
// Nếu tồn tại sách A, thì tăng lên 1
}
else
    $_SESSION['cart'][$isbn] = 1;
//Ngược lại thêm vào sách A
}
// Code hiển thị giỏ hàng ở đây
```

1 loại script có thể dùng chung cho cả edit va add:

Ví dụ :

Mình muốn edit user hay add user :

Edit_&_add.php

PHP Code:

```
<?php
function edit_&_add($user='')
{
    $edit = is_array($user);
//Khi sửa thì chúng ta cần truyền 1 mảng nhận được từ $result = mysql_fetch_assoc($query)
//vào, vì khi ấy mới có đầy đủ thông tin để xuất ra form .
//Nếu $edit là true thì form này dùng để sửa
// Nếu $edit là false thì form này dùng để add
?>
    <form action = <?php echo $edit ? 'edit_user.php' : 'add_user.php' ; ?> method = 'post' >
    <table>
```

```

        <tr><td>Username: </td>
        <td><input type= 'text' name = 'username' value= "<?php echo $e
dit ? $user['username'] : ' ' ;?> " /></td>
        </tr>
        <tr>
        <td>Password: </td>
        <td><input type= 'password' name = 'password' /></td>
        </tr>
    </table>
</form>

```

Ở trên mình đã hướng dẫn sơ bộ quy trình 1 trang bán sách đơn giản . Nếu các bạn cảm thấy phức tạp, thì có thể bỏ 1 vài tính năng mà mình nêu trên, bỏ 1 vài file, v.v .

Quan trọng là hiểu được nó run như thế nào, còn code thì mình nghĩ hok khó lắm đâu .

Các bạn cố gắng làm nhé ;) .

Hướng dẫn chi tiết code trang bán sách

I/ Tạo database :

Bạn tạo database tên là books (hoặc tên bạn muốn) rồi tạo các table sau:

create table customers

```

(
customerid int unsigned not null auto_increment primary key,
name char(60) not null,
address char(80) not null,
city char(30) not null,
state char(20),
zip char(10),
country char(20) not null
);

```

Giải thích và 1 số lưu ý:

Unsigned phải đứng trước not null

Unsigned : hok được âm

Not null : hok được bỏ trống

Char(x): kiểu dữ liệu text tối đa x kí tự

Auto_increment : tự động tăng

create table orders

```
(  
orderid int unsigned not null auto_increment primary key,  
customerid int unsigned not null,  
amount float(6,2),  
date date not null,  
confirm char(1) not null  
);
```

Giải thích :

Int : dữ liệu số

Float(6,2) : phần nguyên tối đa là 6 chữ số, phần thập phân tối đa 2 chữ số

Date: kiểu dữ liệu ngày tháng

create table books

```
(  
isbn char(13) not null primary key,  
author char(80),  
title char(255),  
cate_id int unsigned,  
price float(4,2) not null,  
description longtext  
);
```

Giải thích :

Isbn là đặc trưng của 1 cuốn sách, nó luôn khác nhau nên ta hok cần cột book_id mà chỉ định luôn isbn là primary key

Longtext : kiểu dữ liệu text dung lượng lớn

create table categories

```
(  
ca_tid int unsigned not null auto_increment primary key,  
cate_name char(60) not null  
);
```

create table order_items

```
(  
orderid int unsigned not null,
```

```
isbn char(13) not null,  
item_price float(4,2) not null,  
quantity tinyint unsigned not null,  
primary key (orderid, isbn)  
);
```

create table user

```
(  
user_id int unsigned not null auto_increment primary key,  
username char(16) not null,  
password char(40) not null,  
level tinyint unsigned not null  
);
```

Giải thích:

Tinyint : kiểu dữ liệu số dung lượng nhỏ

LUU Ý : Các bạn bấm Ctrl+S save file php tiếng việt, rồi hiện ra cái bảng, nhấn No, rồi chọn save has utf-8 nhé ;)

II/ Xây dựng trang login

Đầu tiên chúng ta xây dựng trang quản trị với các chức năng được đã được liệt kê ở bài post trước . Tạo folder 'admin' với các file :

Admin.php

Delete_book.php

Delete_cate.php

Delete_user.php

Edit_book.php

Edit_cate.php

Edit_user.php

Insert_book.php

Insert_cate.php

List_cates.php

List_users.php

Tạo 1 trang **login.php** ngang cấp với folder admin . Trang này dùng để đăng nhập :
PHP Code:

```
<?php  
    session_start();  
    if(isset($_SESSION['cart']) && $_SESSION['total_prices'] !=0 ){  
        header('location: show_cart.php');  
    }  
    if(isset($_POST['ok'])){  
        $name = $pass = '';
```

```

        if($_POST['txtname'] == NULL){
            echo 'Bạn chưa nhập username<br/>' ;
        }

        else
            $name = $_POST['txtname'];
        if($_POST['txtpass'] == NULL){
            echo 'Bạn chưa nhập password<br/>' ;
        }
        else
            $pass = $_POST['txtpass'];

        if($name && $pass ) {
            $conn = mysql_connect('localhost', 'root', '');
            mysql_select_db('books');
            $sql = "select * from user where username = '$name' and pas
sword = '$pass' ";
            $result = mysql_query($sql);
            if(!$result){
                echo ' Có lỗi , không thể query database được' ;
            }
            else{
                $array = mysql_fetch_assoc($result);
                $rows = mysql_num_rows($result);

                if($rows == 0){
                    echo 'Bạn đã nhập sai password hoặc username';
                }
                else{
                    session_start();

                    $_SESSION['user'] = $name;
                    $_SESSION['user_id'] = $array['user_id'];
                    $_SESSION['level'] = $array['level'];
                    if($array['level'] ==2){
                        // Nếu người đang truy cập là admin thì chuyển sang trang admin.php
                        header("Location: admin/admin.php");
                        exit();
                    }
                    else{
                        //Ngược lại chuyển sang trang index.php
                        header('Location: index.php');
                        exit();
                    }
                }
            }
        }
    }
}
?>

<form action = 'login.php' method = 'post' >
    <fieldset>
        <legend> Log-in Form </legend>
        <table>
            <tr>
                <td> Name : </td>
                <td> <input type = 'text' name = 'txtname' size = '25' /></td>
            </tr>
        </table>
    </fieldset>
</form>

```

```

        </tr>
        <tr>
        <td> Password : </td>
        <td> <input type = 'password' name = 'txtpass' size = '25' /> </td>
        </tr>
        <tr>
        <td> </td>
        <td> <input type = 'submit' name = 'ok' value = 'Log in' /> </td>
    <!--
    Đặt tên cho field submit để dễ xác nhận xem người dùng có nhấn Submit
    chưa -->
    </tr>
    </table>
    </fieldset>
</form>

```

Giải thích :

Chúng ta quy định : nếu muốn đang mua hàng mà đăng nhập vào trang admin thì hok cho, chuyển người dùng về giỏ hàng, nếu muốn đăng nhập thì set giỏ hàng la 0 hết để bỏ sản phẩm ra khỏi giỏ hoặc tiến tới thanh toán rồi mới cho đăng nhập .

Ở đây dùng form và cách xử lí form trên cùng 1 trang nên chúng ta phải có cách để xác nhận là nút Submit của form đã được nhấn chưa ? Nếu đã nhấn rồi thì mình xử lí form, còn nếu không thì hiển thị form cho người dùng nhập .

Để giải quyết điều đó chúng ta có :

PHP Code:

```

if(isset($_POST['ok'])) {
    //Xử lí form
}

```

Ở PHP 5, để sử dụng biến, chúng ta phải khai báo trước, giả sử như form không được nhập gì hết mà nhấn submit

=> \$name = \$pass = ""; //Set giá trị là NULL

Để xác định xem người dùng có nhập vào chưa, nếu chưa nhập thì báo lỗi :

PHP Code:

```

if($_POST['txtname'] == NULL)
{
    //Báo lỗi
}

```

Sau đó nếu đã tồn tại giá trị \$name và \$pass thì tiến hành xử lí form .

Chúng ta có các dòng code kết nối database rất thường xuyên sử dụng sau :

PHP Code:

```

$conn = mysql_connect('localhost', 'root', '');
mysql_select_db('books');

```

Vì thế chúng ta sẽ tạo 1 file tên connect.php , rồi copy đoạn code trên vào file connect.php trong folder includes ngang cấp với folder admin (nhớ bọc bằng <?php ?>) để sau này tiện dùng lại .

Biến \$result là để lưu kết quả của việc thực hiện truy vấn tới database, nó sẽ trả về TRUE nếu thực hiện được và FALSE nếu xảy ra lỗi trong câu QUERY . Điều này cũng không cần thiết lắm chỉ là khi xảy ra lỗi thì bạn biết lỗi ở chỗ nào . Có thể bỏ qua

Ở trên chúng ta đã xem xét username và password người dùng nhập vào có đúng hay sai , nếu đúng thì cấp 1 phiên làm việc cho người đó với 3 biến \$_SESSION để sau này tiện dùng lại .

II/ Xây dựng trang quản trị

a/ admin.php

Với trang admin.php , chúng ta cần tạo 1 menu cho admin lựa chọn :

PHP Code:

```
<?php
    require_once('../includes/session.php');
    // Kí hiệu ../ có nghĩa là back lại 1 cấp, file này đang ở folder admin
    , nhưng file session.php lại ở ngang cấp với folder admin
?>
    <link href = '../styles.css' rel = 'stylesheet' type = 'text/css' /
>

<table border = '1' cellspacing = '0'>
<tr>
    <th class = 'green_th' colspan = '6' align = 'center' > Welcome <?
php echo "<font color = 'red'> Admin </font>".$_SESSION['user'] ?> </th
>
</tr>
<tr>
    <td class = 'green_td'><a href = '../index.php'> Đến trang chủ </a>
</td>
    <td class = 'green_td'><a href = 'insert_cate.php'> Thêm chuyên mục
</a> </td>
    <td class = 'green_td'><a href = 'insert_book.php'> Thêm sách </a>
</td>
    <td class = 'green_td'><a href = 'list_users.php'> Hiện tài khoản
</a> </td>
    <td class = 'green_td'><a href = 'list_cates.php'> Hiện các chuyên
mục </a> </td>
    <td class = 'green_td'><a href = '../logout.php'> Log-
out </a> </td>
</tr>
</table>
```

Ở trên dòng :

Mã:

```
<link href = '../styles.css' rel = 'stylesheet' type = 'text/css' />
```

: để sử dụng file CSS .

Các class = 'green_td' chỉ là để trang trí bằng CSS, các bạn đừng quan tâm nó !

Như mình đã nói, tất cả các file trong admin, cần phải kiểm tra xem có phải admin đang truy cập không ? Nếu không thì chuyển ngay về trang login.php.

Vì dùng rất nhiều nên chúng ta để các dòng code này vào file session.php trong folder includes ngang cấp với folder admin:

PHP Code:

```
<?php
    session_start();
    if($_SESSION['level'] != 2){
        header('Location: ../login.php');
        exit();
    }
?>
```

Mục đích của trang admin.php này là để tạo 1 menu cho admin lựa chọn . Với mỗi lựa chọn của admin, chuyển admin đến trang tương ứng .

Ở đây có các đường dẫn đến :

Đến trang chủ, thêm chuyên mục, thêm sách, hiện các users, hiện các chuyên mục, và log-out .

II/Xây dựng trang quản trị :

b/ Thêm chuyên mục :

Tiếp theo, mình sẽ xử lý file thêm chuyên mục : insert_cate.php

PHP Code:

```
<?php
    require_once('../includes/session.php');
    require_once('../includes/check_data.php');
    if(isset($_POST['ok'])){
        $cate_name = '';
        if(!check_data($_POST)){
            echo 'Bạn chưa điền hết thông tin của sách';
        }
        else{
            require_once('../includes/connect.php');

            $cate_name = trim(addslashes($_POST['category']));

            $sql = "insert into categories(cate_name)
            values('$cate_name') ";

            $result = mysql_query($sql);
            if($result){
                header("location: list_cates.php");
                exit();
            }
            else
                echo ' Xảy ra lỗi , chuyên mục chưa được thêm vào ' ;
        }
    }
?>

<form action='insert_cate.php'      method = 'post' >
    <fieldset>
```

```

<legend> Thông tin chuyên mục : </legend>
<table>
<tr>
<td>Tên chuyên mục: </td>
<td><input type = 'text' name = 'category' /> </td>
</tr>

<tr>
<td align = 'center' colspan = '2' ><input type = 'submit' name
= 'ok' value = 'Thêm' /></td>
</tr>
</table>
</fieldset>
</form>

```

Thay vì cứ mỗi field trong form, chúng ta check xem nó có NULL hay không , chúng ta cần dùng 1 hàm để xử lý chuyện đó và bỏ nó vào 1 file trong folder includes :

check_data.php

PHP Code:

```

<?php
function check_data($vars){
    foreach ($vars as $key => $value){
        if($key != 'image'){

            if ( !isset($key) || $value == '' ){
                return false;
            }
        }
    }
    return true;
}
?>

```

Hàm này sẽ được truyền như sau : check_data(\$_POST)

Thực ra, POST là 1 mảng, có các khóa là tên của các field và giá trị là thông tin của field đó được input vào . Vì thế chúng ta dùng foreach(\$vars as \$key => \$value) để xử lý theo từng field .

Nếu không tồn tại key hoặc giá trị của field là NULL thì hàm trả về false .

Ở trên, sách có thể có hình hay không hình, vì thế field image có thể là NULL , vậy chúng ta chỉ kiểm tra các field khác của form :
If(!\$key != 'image')

Ở đây chúng ta cần phải addslashes và trim giá trị của các field như :
cate_name . Vì ví dụ như khi nhập :

Cate_name : Mac's . << Với giá trị có “ ‘ “, câu truy vấn sẽ không thực hiện được và trả về false . Vì thế chúng ta phải thêm \ vào trước ‘ trước khi

chuyển vào database .

Quan trọng là, nếu người dùng nhập title :

Mac's : vậy rất phí tài nguyên và sẽ gây khó khăn về sau , vì thế chúng ta phải bỏ các khoảng trắng ở 2 bên chúng, tức là dùng hàm trim()..

c/ Thêm sách :

Tiếp theo, mình sẽ xử lý file thêm sách : **insert_book.php**

PHP Code:

```
<?php

require_once('../includes/session.php');
require_once('../includes/check_data.php');
if(isset($_POST['ok'])){
    $isbn = $author = $title = $price = $description = '';
    if(!check_data($_POST)){
        echo 'Bạn chưa điền hết thông tin của sách';
    }
    else{
        require_once('../includes/connect.php');

        $isbn    = trim($_POST['isbn']);
        $author   = trim(addslashes($_POST['author']));
        $cate_id  = $_POST['cate_id'];
        $title    = trim(addslashes($_POST['title']));
        $price    = $_POST['price'];
        $description = addslashes($_POST['description']);
        if($_FILES['image']['name']){
            move_uploaded_file($_FILES['image']['tmp_name'], "images
/". $_FILES['image']['name']);
            $image = addslashes($_FILES['image']['name']);
            $sql = "insert into books(isbn,author,title,cate_id,ima
ge,price,description)
                values('$isbn','$author','$title', '$cate_id', '$image'
, '$price', '$description')";
        }
        else
            $sql = "insert into books(isbn,author,title,cate_id,pri
ce,description)
                values('$isbn','$author','$title', '$cate_id', '$price',
'$description')";

        $result = mysql_query($sql);
        if($result){
            header("location: ../index.php?cate_id=$cate_id");
            exit();
        }
        else
            echo ' Xảy ra lỗi , sách chưa được thêm vào ' ;
    }
}
```

```

    }
}
?>
<form action = 'insert_book.php'      method = 'post' enctype = 'multipart/form-data'>
    <fieldset>
    <legend> Thông tin sách : </legend>
    <table>
    <tr>
    <td>Chuyên mục: </td>
    <td>
        <select name = 'cate_id'>
            <?php
                require_once('../includes/connect.php');
                $sql = 'select * from categories';
                $result = mysql_query($sql);
                while($rows = mysql_fetch_assoc($result))
//Dùng mysql_fetch_assoc sẽ tiết kiệm bộ nhớ hơn mysql_fetch_array
                echo "<option value = '$rows[cate_id]'">$rows[cate_name] </option>";
//Tại sao hok phải là $result['cate_id'] mà là $result[cate_id] <<< Vì
được lồng vào " " của dòng ECHO
            ?>
        </td>
    </tr>
    <tr>
    <td>ISBN: </td>
    <td><input type = 'text' name = 'isbn' /> </td>
    </tr>
    <tr>
    <td>Tác giả: </td>
    <td><input type = 'text' name = 'author' /> </td>
    </tr>
    <tr>
    <td>Tựa sách: </td>
    <td><input type = 'text' name = 'title' /> </td>
    </tr>
    <tr>
    <td>Hình đại diện: </td>
    <td><input type = 'file' name = 'image' /> </td>
    </tr>
    <tr>
    <td>Giá sách: </td>
    <td><input type = 'text' name = 'price' /> </td>
    </tr>
    <tr>
    <td>Mô tả: </td>
    <td><textarea name = 'description' cols = '50' rows = '20'></textarea></td>
    </tr>
    </fieldset>
    <tr>
    <td colspan = '2' align = 'center'><input type = 'submit' name = 'ok' value = 'Send' /> </td>

```

```
</tr>
</form>
```

Giải thích vòng lặp while :

PHP Code:

```
while($rows = mysql_fetch_assoc($result))
//Dùng mysql_fetch_assoc sẽ tiết kiệm bộ nhớ hơn mysql_fetch_array
    echo "<option value = '$rows[cate_id] '$rows[cate_n
ame] </option>";
```

Ở đây, \$result là kết quả trả về của câu truy vấn, nó sẽ có kiểu dữ liệu là resource, để dễ thao tác, chúng ta chuyển nó thành 1 mảng hok tuần tự với các khóa là tên các cột, giá trị là giá trị của cột đó . Vì kết quả trả về có thể là nhiều dòng, nên chúng ta cần vòng lặp while để duyệt từng dòng :

PHP Code:

```
while($rows = mysql_fetch_assoc($result))
```

Với mỗi dòng duyệt qua, gán mảng bất tuần tự của dòng đó cho biến \$rows, rồi chuyển con trỏ mảng lên dòng kế tiếp

Ở phần thêm sách này, chúng ta cho phép người dùng upload file, nên khi khai báo form phải có :

Mã:

```
enctype = 'multipart/form-data'
```

Vì thêm ảnh đại diện cho sách là 1 field lựa chọn, tức là tùy ý người dùng, nên ta phải chia thành 2 trường hợp, nếu người dùng đã upload ảnh, và khi người dùng không upload ảnh ! Với mỗi trường hợp ta gán câu truy vấn vào biến \$sql .

Riêng với trường hợp upload ảnh , chúng ta cần chuyển file ảnh vào folder images (trong folder admin) .

d/ Hiện các chuyên mục :

Tiếp theo, mình sẽ xử lí file list chuyên mục : [list_cates.php](#)

PHP Code:

```
<link href = '../styles.css' rel = 'stylesheet' type = 'text/css' />
<script>
    function xacnhan(){
        if(!window.confirm("Bạn có thực sự muốn xóa chuyên mục này ?"))
        {
            return false;
        }
    }
</script>
<?php
    require_once('../includes/session.php');
    require_once('../includes/connect.php');
?>
    <table border = '1' cellspacing = '0'>
```

```

<tr>
<th class = 'green_th'> Số thứ tự </th>
<th class = 'green_th'>Tên chuyên mục </th>
<th class = 'green_th'> Số sách trong chuyên mục </td>
<th class = 'green_th' colspan = '2'> Một số quyền hạn </th>
</tr>

<?php
    $sql = 'select * from categories ';
    $result = mysql_query($sql);
    $stt =0;
    while($rows = mysql_fetch_assoc($result)){
        $count_book_query = "select isbn from books where cate_
id =$rows[cate_id]";
        $result_count = mysql_query($count_book_query);
        $count = mysql_num_rows($result_count);
        $stt++;
        echo " <tr>
            <td class = 'green_td' align = 'center'>$stt </td>
            <td class = 'green_td' align = 'center'>$rows[cate_
name]</td>
            <td class = 'green_td' align = 'center'>$count</td>
            <td class = 'green_td' align = 'center'><a href = \
"edit_cate.php?cate_id=$rows[cate_id]"> Edit </a> </td>
            <td class = 'green_td' align = 'center'><a href = \
"delete_cate.php?cate_id=$rows[cate_id]" onclick = 'return xacnhan();'
> Delete </a></td>
            </tr>";
    }
?>
</table>
<?php
    echo "<a href = '../logout.php'> Log out | </a>";
    echo "<a href = 'admin.php'> Về trang quản trị </a>";
?>

```

Tại đây chúng ta tạo 1 table với các cột : số thứ tự, tên chuyên mục, số sách trong chuyên mục, và 1 số quyền hạn của admin như sửa, xóa

Để hiện ra list chuyên mục trong database, chúng ta cần truy vấn rồi dùng vòng lặp để đổ dữ liệu ra bảng

Để làm được điều này , chúng ta cần dùng vòng lặp while :

PHP Code:

```

while($rows = mysql_fetch_assoc($result)){
    $count_book_query = "select isbn from books where cate_
id =$rows[cate_id]";
    $result_count = mysql_query($count_book_query);
    $count = mysql_num_rows($result_count);
    $stt++;
    echo " <tr>
        <td class = 'green_td' align = 'center'>$stt </td>
        <td class = 'green_td' align = 'center'>$rows[cate_
name]</td>
        <td class = 'green_td' align = 'center'>$count</td>
        <td class = 'green_td' align = 'center'><a href = \

```

```

"edit_cate.php?cate_id=$rows[cate_id]\"> Edit </a> </td>
    <td class = 'green_td' align = 'center'><a href = \
"delete_cate.php?cate_id=$rows[cate_id]\" onclick = 'return xacnhan();'
> Delete </a></td>
    </tr>";
}

```

Với mỗi chuyên mục, chúng ta thực hiện 1 câu truy vấn, để lấy dữ liệu của chuyên mục đó. Ngay sau đó dùng `mysql_fetch_assoc()` để chuyển hóa dữ liệu đó thành mảng không tuần tự rồi tiến hành đổ ra.

Với mỗi chuyên mục cho phép admin xóa và sửa, khi click vào thì sẽ chuyển qua trang `delete_cate.php` hoặc `delete_book.php` và kèm theo id của chuyên mục đó:

PHP Code:

```

delete_cate.php?cate_id=$rows[cate_id]
Edit_book.php?cate_id=$rows[cate_id]

```

Lưu ý: chúng ta để phòng trường hợp admin lỡ bấm vào nút xóa, vì thế cần đoạn javascript để hiện ra 1 cái bảng confirm:

Mã:

```

<script>
    function xacnhan(){
        if(!window.confirm("Bạn có thực sự muốn xóa chuyên mục
này ?")){
            return false;
        }
    }
</script>

```

Sử dụng:

Mã:

```

<td class = 'green_td' align = 'center'><a href = \"delete_cate.php?
cate_id=$rows[cate_id]\" onclick = 'return xacnhan();'> Delete </a></td>

```

Giải thích: nếu bạn nhấp ‘No’ thì không có gì xảy ra, nếu nhấp ‘Yes’ thì chuyển đến trang `delete_cate.php?cate_id=...` để thực hiện việc xóa chuyên mục

e/ Xóa chuyên mục

Chúng ta thực hiện xóa chuyên mục như sau: `delete_cate.php`

PHP Code:

```

<?php
    require_once('../includes/session.php');
    require_once('../includes/connect.php');

    $cate_id = $_GET['cate_id'];
    $sql = "delete from categories where cate_id = '$cate_id'";
    $result = mysql_query($sql);
    if($result){
        header('Location : list_cates.php');
        exit();
    }
    else
        echo ' Xảy ra lỗi, chuyên mục chưa được xóa ';
?>

```

Nếu xóa được, chuyển admin sang trang liệt kê chuyên mục . Nếu không, báo lỗi

f/ Sửa chuyên mục :

Và trang sửa chuyên mục **edit_cate.php** như sau:

PHP Code:

```
<?php
    require_once('../includes/session.php');
    require_once('../includes/connect.php');
    require_once('../includes/check_data.php');

    $cate_id = $_GET['cate_id'];
    $sql = " select * from categories where cate_id = '$cate_id' ";
    $result = mysql_query($sql);
    $row = mysql_fetch_assoc($result);

    if(isset($_POST['ok'])){
        if(!check_data($_POST)){
            echo ' Bạn chưa điền đủ thông tin ';
        }
        else{
            $cate_name = $_POST['category'];

            $sql = "update categories set cate_name = '$cate_name'
where cate_id = '$cate_id' ";
            $result = mysql_query($sql);
            if($result){
                header('Location: list_cates.php');
                exit();
            }
            else
                echo ' Xảy ra lỗi, chưa thể thay đổi thông tin chuyên mục';
        }
    }
?>

<form action ='edit_cate.php?cate_id=<?php echo $cate_id; ?>' method = 'post' >
    <fieldset>
        <legend> Thông tin chuyên mục : </legend>
        <table>
            <tr>
                <td>Tên chuyên mục: </td>
                <td><input type ='text' name = 'category' value = "
<?php
                    echo "$row[cate_name]";
//Chú ý không để khoảng trắng trong " " vì lúc đó database sẽ lưu luôn
?> "

            </td>
        </tr>

        <tr>
            <td align ='center' colspan = '2' ><input type = 'submit' name
```

```
= 'ok' value = 'Thêm' /></td>
    </table>
    </fieldset>
</form>
```

Ở file này, chúng ta có các công việc sau :

Hiển thị form cho admin sửa và tại field cate_name của form, phải đổ dữ liệu từ database ra . (Vd chuyên mục kinh tế thì tại field cate_name phải hiện chữ kinh tế) .

Để làm điều này chúng ta dùng thuộc tính value = ' content ' của <input />

Cụ thể là:

Mã:

```
<input type = 'text' name = 'category' value = "<?php
    echo "$row[cate_name]";
?> "
```

Ở trên chúng ta lưu ý không được để khoảng trống hay enter ở trong “ “ của value . Vì mỗi khoảng trắng trong đó sẽ được tính là 1 ký tự . Đây là 1 lỗi khá dễ mắc !!!!

III/Layout:

a/ Trang chủ

Tiếp theo, mình sẽ xử lí file layout sách : **index.php** . Đặt file này ngang cấp với folder admin

PHP Code:

```
<link href = 'styles.css' rel = 'stylesheet' type = 'text/css' />
<?php
    require_once('header.php');
    require_once('includes/connect.php');
    echo '<h1><font color = "red"> Các chuyên mục : </font></h1> ';
    $sql = 'select * from categories';
    $result = mysql_query($sql);
    echo '<ul>';
    while($rows = mysql_fetch_assoc($result)){
        echo "<li><a href = \"list_books.php?cate_id=$rows[cate_id]\" >
$rows[cate_name] </li>";
    }
    echo '</ul>';
    session_start();
    if(!isset($_SESSION['level'] ))
        echo '<a href = "login.php"> Đăng nhập </a>';

    else if($_SESSION['level'] == '2')
    {

        echo '<a href = "admin/admin.php"> Đến trang quản trị </a>';
        echo '<a href = "logout.php"> Log Out</a>';
    }

    //require_once('footer.php');
?>
```

Tại trang này, chúng ta sẽ list các chuyên mục theo dạng unordered list() có trong database, nhưng không phải để sửa, xóa như bên admin mà là để xem sách trong chuyên mục đó .

Để làm được điều này, chúng ta chỉ cần thực hiện 1 câu truy vấn để lấy tất tần tật thông tin về chuyên mục rồi chuyển hóa nó thành mảng . Sau đó xuất ra bằng dòng while . Tại mỗi chuyên mục, đặt 1 link đến file **list_books.php?cate_id=...** <<<**list_books.php** có nhiệm vụ liệt kê sách trong chuyên mục với cate_id tương ứng .

Tại đây, nếu người đang truy cập là admin thì hiện chữ : Về trang quản trị và log-out

Nếu không phải thì hiện chữ : đăng nhập

b/ Hiện các sách có trong chuyên mục :

Tại file **list_books.php** :

PHP Code:

```
<link href = 'styles.css' rel = 'stylesheet' type = 'text/css' />
<?php
    require_once('header.php');
    require_once('includes/connect.php');
    session_start();

    $cate_id = $_GET['cate_id'];
    $sql = "select isbn,image,title from books where cate_id = '$cate_id' ";
    $result = mysql_query($sql);
    echo '<table>';
    while($rows = mysql_fetch_assoc($result)){
        if($rows['image'] != NULL){
            echo "<tr>
                <td><img src = \"admin/images/$rows[image]\" width = '100px' height = '150px' border = '1' /></td>
                <td><a href = \"book_info.php?isbn=$rows[isbn]\" > $rows[title] </li>";
        }
        else
            echo "<tr>
                <td><img src = # width = '100px' height = '150px' /></td>
                <td valign = 'center'><a href = \"book_info.php?isbn=$rows[isbn]\" > $rows[title] </li>";
    }
    echo '</table>';
    echo '<a href = "index.php"> Đến các chuyên mục khác </a>';

    if(!isset($_SESSION['level']) || $_SESSION['level'] != '2')
        echo '<a href = "show_cart.php"> Xem giỏ hàng của bạn </a>';

    else
        echo '<a href = "logout.php"> Log Out</a>';
```

```
//require_once('footer.php');
?>
```

Chúng ta sẽ liệt kê các sách trong chuyên mục theo 1 bảng, cột bên trái là hình đại diện, cột bên phải là tựa sách, tựa sách có gắn link đến file **book_info.php?isbn...** <<<file **book_info.php** có nhiệm vụ liệt kê thông tin sách có isbn chỉ định

Vì hình ảnh là optional, vì thế chúng ta phải xét 2 trường hợp, nếu hình trên database là NULL, thì chúng ta sẽ không show hình ra (src = #), ngược lại, trở đường dẫn tới folder images đã tạo, lấy hình đã upload show ra : **admin/images/\$rows[image]**

c/ Hiển thị thông tin sách :

PHP Code:

```
<link href = 'styles.css' rel = 'stylesheet' type = 'text/css' />
<script>
function xacnhan()
{
    if(!window.confirm("Bạn thực sự muốn xóa sách này? " )){
        return false;
    }
}
</script>

<?php
require_once('header.php');
require_once('includes/connect.php');
session_start();

$isbn = $_GET['isbn'];
$sql = "select * from books where isbn = '$isbn' ";
$result = mysql_query($sql);
$rows = mysql_fetch_assoc($result);
$isbn = htmlspecialchars($rows['isbn']);
$title = htmlspecialchars($rows['title']);
$author =htmlspecialchars($rows['author']);
$description = htmlspecialchars($rows['description']);
echo '<table>';

    if($rows['image'] != NULL){
        echo "<tr>
                <td><img src = \"admin/images/$rows[image]\" wid
th = '100px' height = '150px' /></td>";

        }
    else
        echo "<tr>
                <td><img src = # width = '100px' height = '150px'
/></td>";
        echo "<td><ul>
                <li>ISBN: $isbn</li>
                <li>Title: $title</li>
                <li>Author: $author</li>
                <li>Price: $rows[price]</li>
```

```

                <li>Description: $description </li>
            </ul>";
            echo "</td></tr>";

            echo '</table>';
            if(isset($_SESSION['level']) && $_SESSION['level'] ==2)
            {
                echo "<a href = \"admin/edit_book.php?isbn=$isbn\"> Sửa thông tin sách| </a>";
                echo "<a href = \"admin/delete_book.php?isbn=$isbn\" onclick = \"return xacnhan();\"> Xóa sách khỏi chuyên mục|</a>";
                echo "<a href = \"list_books.php?cate_id=$rows[cate_id]\"> Về trang trước |</a>";
                echo '<a href = "logout.php"> Log Out</a>';
            }
            else{

                echo "<a href =\"list_books.php?cate_id=$rows[cate_id]\"> Tiếp tục mua sách </a>";
                echo "<a href = \"show_cart.php?isbn=$rows[isbn]\"> Thêm vào giỏ hàng| </a>";
                if(isset($_SESSION['cart']))
                    echo "<a href = \"show_cart.php\"> Xem giỏ hàng| </a>";
            }
            //require_once('footer.php');
        }
    }
}

```

Ở phần này, nếu người đang truy cập là admin thì xuất ra : Sửa thông tin sách, Xóa sách khỏi chuyên mục, về trang trước, log out .

PHP Code:

```

if(isset($_SESSION['level']) && $_SESSION['level'] ==2)
{
    echo "<a href = \"admin/edit_book.php?isbn=$isbn\"> Sửa thông tin sách| </a>";
    echo "<a href = \"admin/delete_book.php?isbn=$isbn\" onclick = \"return xacnhan();\"> Xóa sách khỏi chuyên mục|</a>";
    echo "<a href = \"list_books.php?cate_id=$rows[cate_id]\"> Về trang trước |</a>";
    echo '<a href = "logout.php"> Log Out</a>';
}

```

Ngược lại thì xuất ra : tiếp tục mua sách, thêm vào giỏ hàng, xem giỏ hàng

PHP Code:

```

else{

    echo "<a href =\"list_books.php?cate_id=$rows[cate_id]\"> Tiếp tục mua sách </a>";
    echo "<a href = \"show_cart.php?isbn=$rows[isbn]\"> Thêm vào giỏ hàng| </a>";
    if(isset($_SESSION['cart']))
        // Nếu đã từng thêm sách vào giỏ hàng thì mới cho xem giỏ hàng, nếu lần đầu truy cập thì hok hiện Xem Giỏ Hàng . Đọc part 9 để hiểu giỏ hàng
        echo "<a href = \"show_cart.php\"> Xem giỏ hàng| </a>";
    }
    //require_once('footer.php');
}

```

III/ Hiển thị giỏ hàng

Trang **show_cart.php** có nhiệm vụ như sau :

Nếu từ file **book_info.php** chúng ta click : thêm vào giỏ hàng => sẽ được chuyển sang show_cart.php?isbn=... <<<Lúc này **show_cart.php** có nhiệm vụ thêm sách đó vào giỏ hàng nếu nó không có và tăng số lượng của nó thêm 1 nếu đã tồn tại !

Nếu từ các file như list_books.php, book_info.php , người dùng click: xem giỏ hàng => sẽ được chuyển sang **show_cart.php** . Tại đây, trang chỉ có nhiệm vụ show ra giỏ hàng cho người truy cập.

Tại trang **show_cart.php** chúng ta cho phép người dùng thay đổi số lượng của sách . Nhấn save lại => Cập nhật tổng tiền, số lượng v.v .

Ta có code trang **show_cart.php** như sau :

PHP Code:

```
<?php
require_once('header.php');
require_once('includes/connect.php');
session_start();
$isbn = '';
if(isset($_GET['isbn'])){
    $isbn = $_GET['isbn'];
    $sql = "select cate_id from books where isbn = '$isbn' ";
    $book_ordered = mysql_query($sql);
    $book_ordered = mysql_fetch_assoc($book_ordered);
    if(!isset($_SESSION['cart'])){
        $_SESSION['cart'] = array();
    }

    if(!isset($_SESSION['cart'][$isbn])){
        $_SESSION['cart'][$isbn] = 1;
    }

    else
        $_SESSION['cart'][$isbn]++;
}

$_SESSION['total_items'] = 0;
$_SESSION['total_prices'] = 0;

if(isset($_POST['save'])){

    foreach($_SESSION['cart'] as $isbn_i => $quantity_q){
        if($_POST[$isbn_i] ==0){
            unset($_SESSION['cart'][$isbn_i]);
        }
        else
            $_SESSION['cart'][$isbn_i] = $_POST[$isbn_i];
    }
}
```


PHP Code:

```
$total_prices = $rows['price']*$quantity_q ;
```

Để tính tổng tiền của tất cả cuốn sách trong giỏ , chúng ta đã tính trước đó :

```
$_SESSION['cart'][$total_prices] .
```

Mà biến `$_SESSION` hok thể nội suy được, vì thế chúng ta gán vào 1 biến tên

```
$total_prices_books
```

Tới phần nút SAVE: Ở đây mình dùng hình ảnh là save.gif làm nút save: cú pháp :

Mã:

```
<input type = 'image' src = 'save.gif' />
```

Nhưng để nút save này có nhiệm vụ như là 1 nút submit chúng ta thêm type = 'hidden' ở trước nó :

Mã:

```
<input type = 'hidden' name = 'save' />
```

Nếu như giỏ hàng mới được bổ sung thêm sách (tức là [show_cart.php?isbn=...](#)) thì 'tiếp tục mua hàng' sẽ dẫn đến chuyên mục chứa sách đó để khách hàng lựa chọn tip

Nếu như giỏ hàng chỉ được xem (tức là [show_cart.php](#)) thì 'tiếp tục mua hàng' sẽ dẫn đến liệt kê các chuyên mục.

PHP Code:

```
if($isbn){
    echo "<a href = \"list_books.php?cate_id=$book_ordered[cate_id]\">
Tiếp tục mua hàng </a>";
}
else
    echo '<a href = "index.php"> Tiếp tục mua hàng </a>';
```

Chúng ta xét trường hợp nếu sách vừa được thêm vào giỏ hàng , tức là trên địa chỉ có isbn :

PHP Code:

```
$isbn = $_GET['isbn'];
//Lấy giá trị của isbn trên thanh địa chỉ
$sql = "select cate_id from books where isbn = '$isbn' ";
//Lấy id của chuyên mục mà sách có isbn đó thuộc
$book_ordered = mysql_query($sql);
//Thực hiện câu truy vấn
$book_ordered = mysql_fetch_assoc($book_ordered);
//Chuyển kết quả trả về thành mảng không tuần tự
if(!isset($_SESSION['cart'])){
    $_SESSION['cart'] = array();
}

if(!isset($_SESSION['cart'][$isbn])){
    $_SESSION['cart'][$isbn] = 1;
}

else
    $_SESSION['cart'][$isbn]++;
}
```

Chúng ta xử lý phần giỏ hàng như sau :

PHP Code:

```
if(!isset($_SESSION['cart'])){
    $_SESSION['cart'] = array();
}

if(!isset($_SESSION['cart'][$isbn])){
    $_SESSION['cart'][$isbn] = 1;
}

else
    $_SESSION['cart'][$isbn]++;
```

Nếu không tồn tại \$_SESSION['cart'] : tức là người dùng chưa mua hàng, thì chúng ta khởi tạo \$_SESSION['cart'] là 1 mảng . Mảng này có nhiệm vụ chứa khóa là isbn, giá trị của khóa là số lượng sách đặt trong giỏ .

Khi ng dùng thêm 1 cuốn sách vào, kiểm tra xem sách đó đã tồn tại chưa ? Nếu chưa thì tạo mới nếu có rồi thì tăng lên 1 .

Sau những công việc trên, chúng ta phải reset giá trị của \$_SESSION['total_prices'] và \$_SESSION['total_items'], rồi thực hiện việc tính lại từ đầu giá trị của 2 SESSION này, nếu không thì phần tính toán sẽ bị sai khi ng dùng save hay refresh .

PHP Code:

```
$_SESSION['total_items'] = 0;
$_SESSION['total_prices'] = 0;
```

Trước khi thực hiện việc tính toán tổng sách và tổng tiền (giá trị của 2 SESSION trên), chúng ta cần xử lí nút save trước :

PHP Code:

```
if(isset($_POST['save'])){
    //Nếu người dùng đã nhấn nút save

    foreach($_SESSION['cart'] as $isbn_i => $quantity_q){
        //Lặp với từng cuốn sách trong giỏ
        if($_POST[$isbn_i] ==0){
            //Nếu số lượng sách của 1 cuốn sách xác định là 0, thì bỏ cuốn sách đó
            ra khỏi giỏ hàng
            unset($_SESSION['cart'][$isbn_i]);
        }
        else
            $_SESSION['cart'][$isbn_i] = $_POST[$isbn_i];
        //Nếu khác không, cập nhật số lượng sách
    }
}
```

Tính toán tổng số sách trong giỏ và tổng tiền :

PHP Code:

```
foreach($_SESSION['cart'] as $isbn_i => $quantity_q){
    $sql = "select price from books where isbn = '$isbn_i' ";
    //Chọn giá của từng cuốn sách từ database
    $result = mysql_query($sql);
    $rows = mysql_fetch_assoc($result);
    //Chuyển về thành mảng không tuần tự
```

```

        $price = $rows['price'];
//Lấy giá của sách gán vào biến

        $_SESSION['total_items'] += $quantity_q;
//Tổng số sách thì được tính dễ dàng bằng cách cộng dồn tổng số lượng t
ùng cuốn sách
        $_SESSION['total_prices'] += $quantity_q*$price;
//Tổng tiền thì cộng dồn tổng tiền mua mỗi loại sách .
    }

```

II/ Trang quản trị:

c/ Sửa thông tin sách

Giờ chúng ta quay lại xử lí trang edit_book.php:

PHP Code:

```

<?php

require_once('../includes/session.php');
require_once('../includes/check_data.php');
require_once('../includes/connect.php');

if(isset($_GET['isbn']))
{
    $isbn = $_GET['isbn'];
    $sql = "select * from books where isbn = '$isbn' ";
}
else if(isset($_POST['old_isbn'])) {
    $old_isbn = $_POST['old_isbn'];
    $sql = "select * from books where isbn = '$old_isbn' ";
}

$book = mysql_query($sql);
$book = mysql_fetch_assoc($book);

if(isset($_POST['ok'])) {
    $isbn = $author = $title = $price = $description = '';
    if(!check_data($_POST)) {
        echo 'Bạn chưa điền hết thông tin của sách';
    }
    else {

        $isbn      = trim($_POST['isbn']);
        $author     = trim(htmlspecialchars($_POST['author']));
        $cate_id    = $_POST['cate_id'];
        $title      = trim(htmlspecialchars($_POST['title']));
        $price      = $_POST['price'];
        $description = trim(htmlspecialchars($_POST['description']))
    );

    if($old_isbn != $isbn) {
        $sql = "select author from books where isbn ='$isbn'";
    }
}

```

```

        $result = mysql_query($sql);
        $rows = mysql_num_rows($result);
        if($rows == 0){
            if($_FILES['image']['name']){
                move_uploaded_file($_FILES['image']['tmp_name'],
"images/".$_FILES['image']['name']);
                $image = $_FILES['image']['name'];
                $sql = "update books set isbn = '$isbn', author
= '$author', title = '$title', cate_id = '$cate_id', image = '$image', p
rice = '$price',
description = '$description' where isbn = '$old_isbn'";
            }
            else
                $sql = "update books set isbn = '$isbn', author
= '$author', title = '$title', cate_id = '$cate_id', price = '$price',
description = '$description' where isbn = '$old_isbn'";

            $result = mysql_query($sql);
        }
        else{
            echo 'Xây ra lỗi .ISBN đã tồn tại';
            $result = false;
        }
    }

    if($old_isbn == $isbn){
        if($_FILES['image']['name']){
            move_uploaded_file($_FILES['image']['tmp_name'],
"images/".$_FILES['image']['name']);
            $image = $_FILES['image']['name'];
            $sql = "update books set isbn = '$isbn', author
= '$author', title = '$title', cate_id = '$cate_id', image = '$image', p
rice = '$price',
description = '$description' where isbn = '$old_isbn'";
        }
        else
            $sql = "update books set isbn = '$isbn', author
= '$author', title = '$title', cate_id = '$cate_id', price = '$price',
description = '$description' where isbn = '$old_isbn'";

        $result = mysql_query($sql);
    }

    if($result){
        header("location: ../list_books.php?cate_id=$cate_id");
        exit();
    }
    else
        echo ' Xây ra lỗi , sách chưa được thêm vào ' ;

}

}

?>

```

```

        <form action = 'edit_book.php'          method = 'post' enctype = 'multi
part/form-data'>
        <fieldset>
        <legend> Thông tin sách : </legend>
        <table>
        <tr>
        <td>Chuyên mục: </td>
        <td>
                <select name = 'cate_id'>
                        <?php
                                require_once('../includes/connect.php');
                                $sql = 'select * from categories';
                                $result = mysql_query($sql);
                                while($rows = mysql_fetch_assoc($result))
//Dùng mysql_fetch_assoc sẽ tiết kiệm bộ nhớ hơn mysql_fetch_array
                                echo "<option value = '$rows[cate_id]'
<?php
        if($rows[cate_id] == $book[cate_id])
                echo 'selected = selected';
        ?>
                                $rows[cate_name] </option>";
//Tại sao hok phải là $result['cate_id'] mà là $result[cate_id] <<< Vì
được lồng vào " " của dòng ECHO
                                ?>
                        </td>
                </tr>
                <tr>
                <td>ISBN: </td>
                <td><input type = 'text' name = 'isbn' value = '<?php
echo htmlspecialchars($book['isbn']);?>' /> </td>
                </tr>
                <tr>
                <td>Tác giả: </td>
                <td><input type = 'text' name = 'author' value = '
<?php
echo htmlspecialchars($book['author']);
?>
                ' /> </td>
                </tr>
                <tr>
                <td>Tên sách: </td>
                <td><input type = 'text' name = 'title' value = '<?php
echo htmlspecialchars($book['title']);
?>' /> </td>
                </tr>
        <?php
                if($book['image'] != ''){
                        echo '<tr>';
                        echo '<td> Hình đại diện cho sách : </td>';
                        echo "<td> <img src = 'images/$book[image]' width = '100px' height
= '150px' /></td>";
                        echo '</tr>';
                }
        ?>

        <tr>

```

```

        <td>Hình đại diện: </td>
        <td><input type = 'file' name = 'image' /> </td>
    </tr>

    <tr>
        <td>Giá sách: </td>
        <td><input type = 'text' name = 'price' value = '
<?php
    echo $book['price'];
?>' /> </td>
    </tr>
    <tr>
        <td>Mô tả: </td>
        <td><textarea name = 'description' cols = '50' rows = '20'>
<?php
    echo htmlspecialchars($book['description']);
?>
        </textarea></td>
    </tr>
</fieldset>
    <tr>
        <td colspan = '2' align = 'center'>
        <input type = 'hidden' name = 'old_isbn' value = "<?php echo $bo
ok['isbn'];?>" />
        <input type = 'submit' name = 'ok' value = 'Send' /> </td>
    </tr>
</form>

```

Ở đây chúng ta có nhiệm vụ đổ dữ liệu từ database ra form . Lưu ý : khi sửa sách, có thể dùng sẽ sửa isbn, mà isbn là primary key trong database . Vậy chúng ta phải xử lý 2 trường hợp:

- 1/ Nếu người dùng thay đổi isbn .
- 2/ Nếu người dùng hok thay đổi isbn

Để xét 2 trường hợp trên, chúng ta cần biết isbn trước khi edit của sách, vì thế chúng ta chèn :

Mã:

```

<input type = 'hidden' name = 'old_isbn' value = "<?php echo
$book['isbn'];?>" />

```

Đoạn code trên sẽ gửi isbn cũ cho chúng ta qua form .

Các bạn thấy, form này gửi dữ liệu đến edit_book.php chứ hok phải edit_book.php?isbn.... như thường lệ . Vậy nếu chúng ta chỉ dùng :

PHP Code:

```

if(isset($_GET['isbn']))
{
    $isbn = $_GET['isbn'];
    $sql = "select * from books where isbn = '$isbn' ";
}

```

Thì khi có lỗi hok thay đổi sách được, PHP sẽ vẫn chạy script edit_book.php mà hok thực hiện việc chuyển trang => Lúc này hok còn isbn nữa để mà \$_GET => form sẽ đầy lỗi vì hok có mảng bắt buộc để đổ dữ liệu từ database

Vì vậy chúng ta thêm :

PHP Code:

```
else if(isset($_POST['old_isbn'])){
    $old_isbn = $_POST['old_isbn'];
    $sql = "select * from books where isbn = '$old_isbn' ";
}
```

Ở đây, cho dù xảy ra sai sót gì đó, thì chúng ta vẫn còn old_isbn để lấy dữ liệu từ database và form sẽ hok còn lỗi :

PHP Code:

```
$book = mysql_query($sql);
$book = mysql_fetch_assoc($book);
if($old_isbn != $isbn){
    $sql = "select author from books where isbn = '$isbn'";
    $result = mysql_query($sql);
    $rows = mysql_num_rows($result);
    if($rows == 0){
        if($_FILES['image']['name']){
            move_uploaded_file($_FILES['image']['tmp_name'],
"images/".$_FILES['image']['name']);
            $image = $_FILES['image']['name'];
            $sql = "update books set isbn = '$isbn', author
= '$author', title = '$title', cate_id = '$cate_id', image = '$image', p
rice = '$price',
description = '$description' where isbn = '$old_isbn'";
        }
        else
            $sql = "update books set isbn = '$isbn', author
= '$author', title = '$title', cate_id = '$cate_id', price = '$price',
description = '$description' where isbn = '$old_isbn'
'";

        $result = mysql_query($sql);
    }
    else{
        echo 'Xây ra lỗi .ISBN đã tồn tại';
        $result = false;
    }
}

if($old_isbn == $isbn){
    if($_FILES['image']['name']){
        move_uploaded_file($_FILES['image']['tmp_name'],
"images/".$_FILES['image']['name']);
        $image = $_FILES['image']['name'];
        $sql = "update books set isbn = '$isbn', author
= '$author', title = '$title', cate_id = '$cate_id', image = '$image', p
rice = '$price',
description = '$description' where isbn = '$old_isbn'";
    }
    else
        $sql = "update books set isbn = '$isbn', author
= '$author', title = '$title', cate_id = '$cate_id', price = '$price',
description = '$description' where isbn = '$old_isbn'";
}
```

```
        $result = mysql_query($sql);  
    }
```

Ở đây. Chúng ta xét 2 trường hợp như đã nói ở trên . Trong mỗi trường hợp , lại có 2 trường hợp con là hok có ảnh đại diện và có ảnh đại diện .

TH : \$old_isbn != \$isbn

Ở đây, chúng ta tìm xem trong database, isbn mới có trùng với isbn của cuốn sách nào khác hay không ?

Để làm điều này ta :

PHP Code:

```
$sql = "select author from books where isbn ='$isbn'";  
$result = mysql_query($sql);
```

Rồi sau đó tìm số dòng của kết quả trả về :

PHP Code:

```
$rows = mysql_num_rows($result);
```

Nếu 0 dòng, tức là không có trùng, chúng ta thực hiện câu truy vấn thay đổi thông tin :

PHP Code:

```
$result = mysql_query($sql) . //trả về true
```

Nêu trùng, chúng ta gán \$result là false;

TH : \$isbn == \$old_isbn

Ở trường hợp này chúng ta làm bình thường, vì isbn được giữ nguyên .

Sau khi xét 2 trường hợp, chúng ta thực hiện:

Nếu result trả về true, thì chuyển đến trang list_cate.php?....

Nếu result trả về false (tức là trường hợp trùng isbn hoặc xảy ra lỗi nào đó) thì ở trang hiện tại .

Vậy là mình đã hướng dẫn các bạn code trang bán sách, hi vọng là các bạn mới học PHP có cái nhìn tổng quan về việc xây dựng 1 ứng dụng đơn giản . Mục đích của những bài hướng dẫn này là như thế, mình chỉ mong những ng mới học, mới bước vào PHP hình dung dc thế nào là code 1 ứng dụng hoàn chỉnh . Vì thế layout mình chưa xây dựng cho đẹp, và code chưa hoàn chỉnh cho lắm . Code này hok dành cho các đối tượng đã hiểu và đã nắm được những điều trên nhé ;;) .

Còn 1 số file khác như edit user, list user,v.v cách giải thích cũng như nhau nên mình hok post lên đây .

Code này còn thiếu phần check out, thanh toán, phần user thì chưa kiểm tra tồn tại username chưa như isbn ở trên , các bạn tự xử lý nhé, coi như bài tập về nhà =]] ^ ^
Chào mọi người nhé !