

# PHP Cơ bản

Giảng viên: TS. Nguyễn Thành Huy



## Nội dung

- Giới thiệu PHP
- Cơ chế hoạt động của WebServer
- Cú pháp & Quy ước trong PHP

## Nội dung

- Giới thiệu PHP
- Cơ chế hoạt động của WebServer
- Cú pháp & Quy ước trong PHP

# Giới thiệu về PHP – Lịch sử phát triển

- **PHP** : [Rasmus Lerdorf](#) in 1994 (được phát triển để phát sinh các form đăng nhập sử dụng giao thức HTTP của Unix)
- **PHP 2 (1995)** : Chuyển sang [ngôn ngữ script xử lý trên server](#). Hỗ trợ CSDL, Upload File, khai báo biến, mảng, hàm đệ quy, câu điều kiện, biểu thức, ...
- **PHP 3 (1998)** : Hỗ trợ ODBC, [đa hệ điều hành](#), giao thức email (SNMP, IMAP), bộ phân tích mã PHP (parser) của [Zeev Suraski](#) và [Andi Gutmans](#)
- **PHP 4 (2000)** : Trở thành một thành phần độc lập cho các webserver. Parse đổi tên thành [Zend Engine](#). Bổ sung các tính năng bảo mật cho PHP
- **PHP 5 (2005)** : Bổ sung Zend Engine II hỗ trợ [lập trình HĐT](#), [XML](#), [SOAP](#) cho Web Services, SQLite
- Phiên bản mới nhất của PHP là version [PHP 5.2.4](#) ([www.php.net](http://www.php.net))



## Giới thiệu về PHP – PHP là gì ?

- **PHP** viết tắt của **P**HP **H**ypertext **P**reprocessor
- Là ngôn ngữ server-side script, tương tự như ASP, JSP, ... thực thi ở phía WebServer
- Tập tin PHP có phần mở rộng là **.php**
- Cú pháp ngôn ngữ giống ngôn ngữ **C & Perl**

**Ưu điểm  
PHP ?**



## Giới thiệu về PHP – Ưu điểm 1

- PHP được sử dụng làm
  - Server Side Scripting
  - CommandLine Scripting (cron - Linux, Task Scheduler - Windows, Text Processing)
  - Xây dựng ứng Desktop - PHP GTK

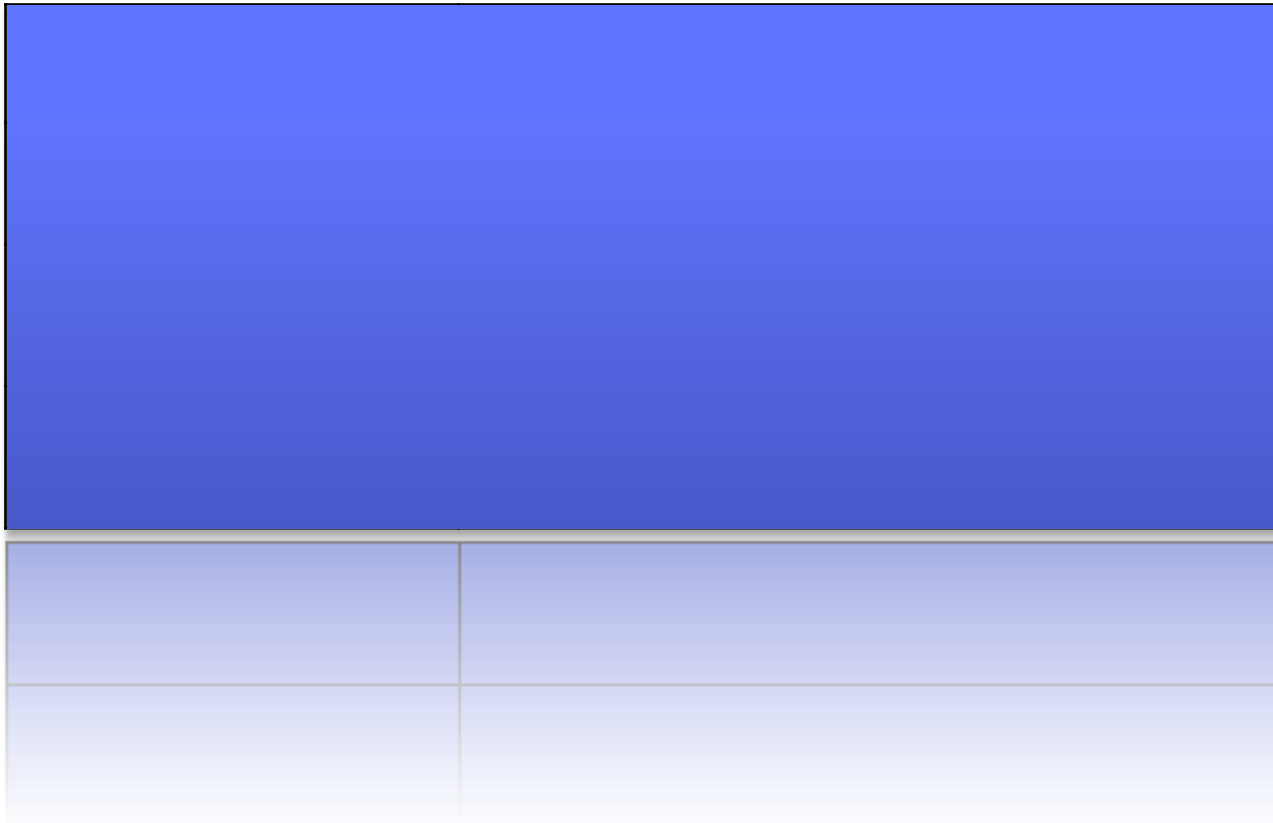
## Giới thiệu về PHP – Ưu điểm 2

### ▪ Đa môi trường (Multi-Platform)

- **Web Servers:** Apache, Microsoft IIS, Caudium, Netscape Enterprise Server
- **Hệ điều hành:** UNIX (HP-UX, OpenBSD, Solaris, Linux), Mac OSX, Windows NT/98/2000/XP/2003/vista
- **Hệ QTCSDL:** Adabas D, dBase, Empress, FilePro (read-only), Hyperwave, IBM DB2, Informix, Ingres, InterBase, FrontBase, mSQL, Direct MS-SQL, MySQL, ODBC, Oracle (OCI7 and OCI8), Ovrimos, PostgreSQL, SQLite, Solid, Sybase, Velocis, Unix dbm

# Giới thiệu về PHP - Ưu điểm 3

- **Miễn phí**

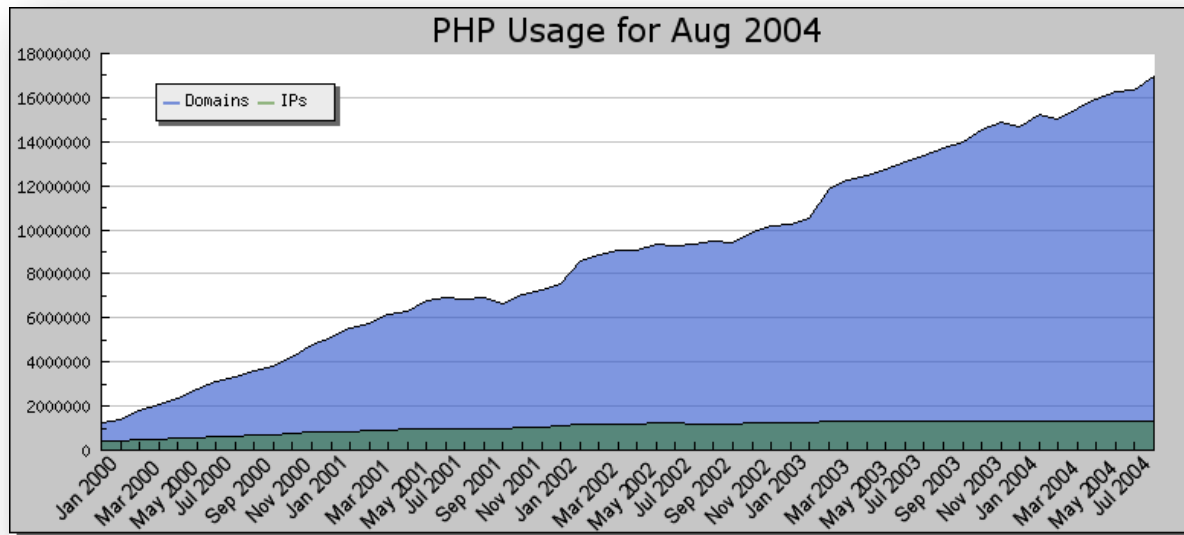




## Giới thiệu về PHP - Ưu điểm 4

- Được sử dụng rộng rãi trong môi trường phát triển web
  - 20,917,850 domains (chiếm hơn 32% tên miền website)
  - 1,224,183 IP addresses

(04/2007 Netcraft Survey - <http://www.php.net/usage.php>)



# Giới thiệu về PHP – Một số website lớn



**PHP at Yahoo!**

<http://www.yahoo.com>

**The Internet's most trafficked site**



*Portal*



*Portal*



*Course Management System*



*Wiki*



*Customer Relationship Management*



*e-Commerce*



*Portal*



*Bulletin Board*



*Content Management System*



*Help Desk*

## Giới thiệu về PHP – Cần gì để chạy PHP ?

### ■ Download PHP

- Download PHP for free here:  
<http://www.php.net/downloads.php>

### ■ Download MySQL Database

- Download MySQL for free here:  
<http://www.mysql.com/downloads/index.html>

### ■ Download Apache Server

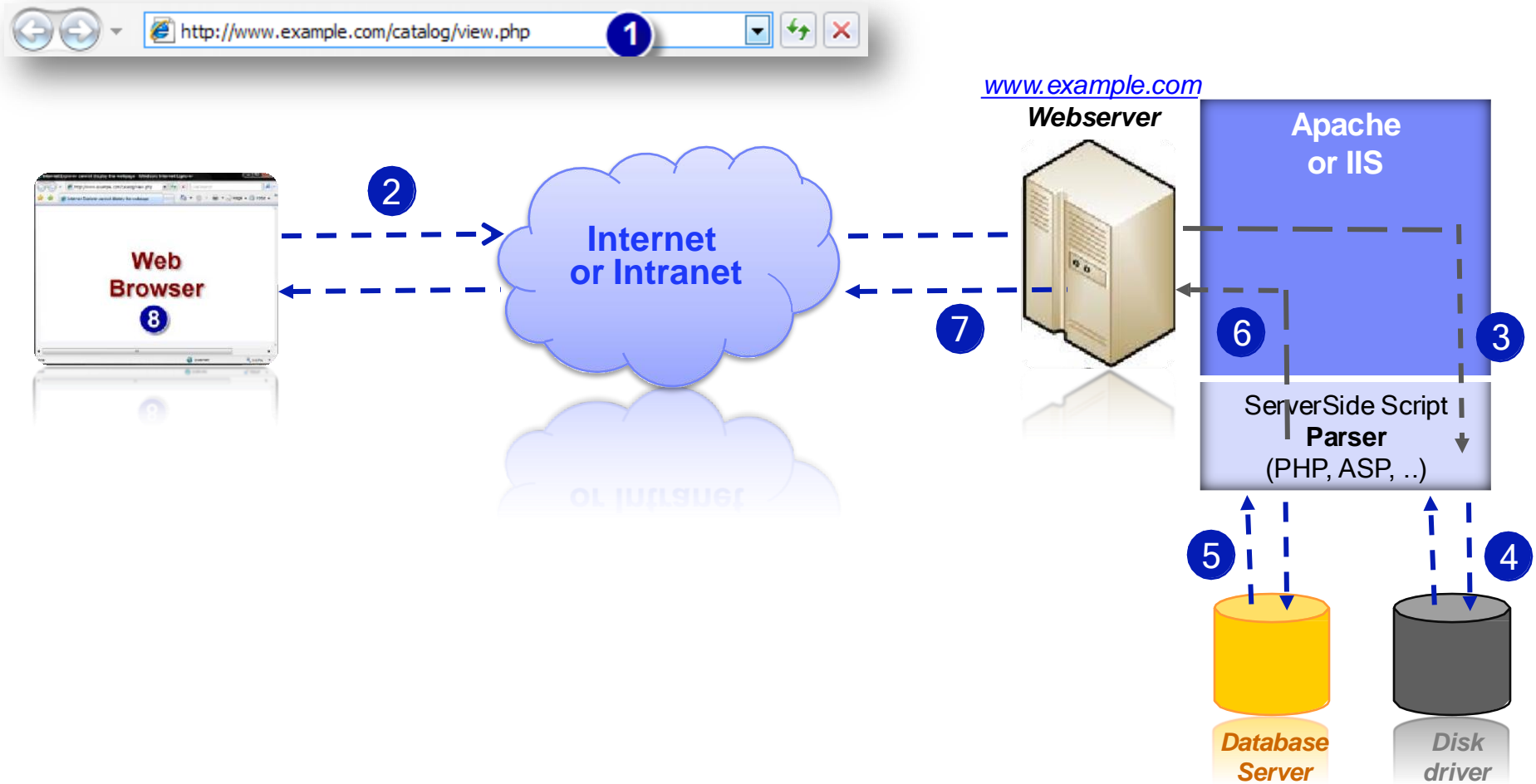
- Download Apache for free here:  
<http://httpd.apache.org/download.cgi>

→ Download **WAMP, LAMP**

## Nội dung

- Giới thiệu PHP
- Cơ chế hoạt động của WebServer
- Cú pháp & Quy ước trong PHP

# Cơ chế hoạt động của WebServer



# Cơ chế hoạt động của WebServer

```
1 <html>
2 <head>
3   <title>Test Server Script Parser</title>
4 </head>
5 <body>
6
7   <h1>Server Script Parser</h1>
8   Hello world HTML
9   <br />
10  <br />
11
12  <% response.write("Hello ASP Parser !!!") %>
13  <br />
14  <br />
15
16  <?php echo "Hello PHP Parser !!!" ?>
17  <br />
18  <br />
19
20 </body>
21 </html>
22
```

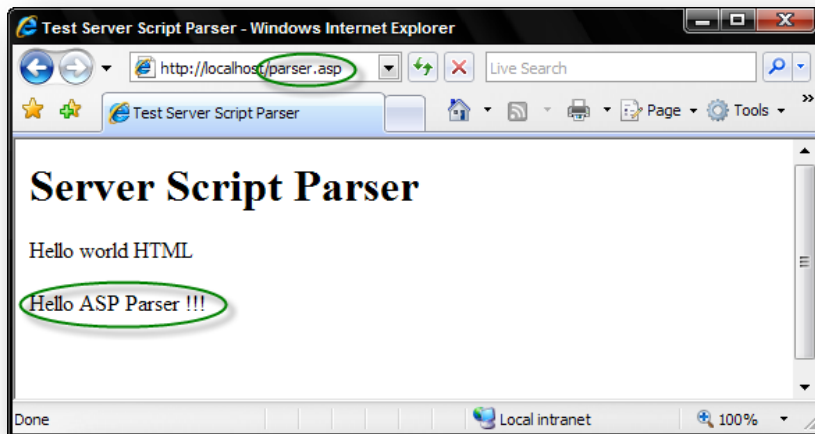
**HTML Code**

**ASP Code**

**PHP Code**

# Cơ chế hoạt động của WebServer

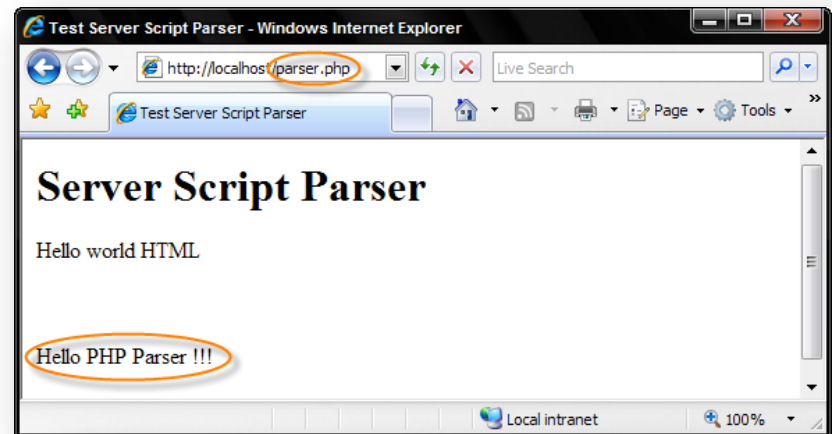
## ■ Parser.asp



```
parser[1] - Notepad
File Edit Format View Help
<html>
<head>
  <title>Test Server Script Parser</title>
</head>
<body>

  <h1>Server Script Parser</h1>
  Hello world HTML
  <br />
  <br />
  Hello ASP Parser !!!
  <br />
  <br />
  <?php echo "Hello PHP Parser !!!" ?>
  <br />
  <br />
</body>
</html>
```

## ■ Parser.php



```
parser[1] - Notepad
File Edit Format View Help
<html>
<head>
  <title>Test Server Script Parser</title>
</head>
<body>

  <h1>Server Script Parser</h1>
  Hello world HTML
  <br />
  <br />
  <% Response.Write("Hello ASP Parser !!!")%>
  <br />
  <br />
  Hello PHP Parser !!! <br />
  <br />
</body>
</html>
```

## Nội dung

- Giới thiệu PHP
- Cơ chế hoạt động của WebServer
- Cú pháp & Quy ước trong PHP



# Cú pháp & Quy ước trong PHP

- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- Toán tử
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng

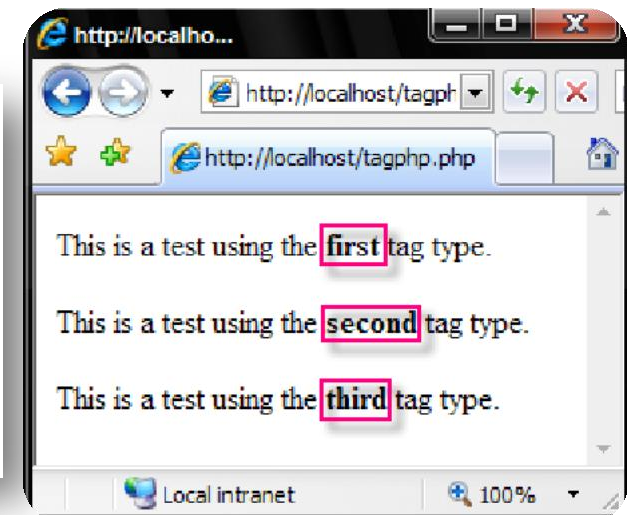
## Quy ước

- Mã lệnh **PHP** được đặt trong các cặp thẻ sau :

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

```

1 <?
2     echo "<P>This is a test using the <b>first</b> tag type.</P>";
3 ?>
4
5 <?php
6     echo "<P>This is a test using the <b>second</b> tag type.</P>";
7 ?>
8
9 <script language="php">
10     echo "<P>This is a test using the <b>third</b> tag type.</P>";
11 </script>
  
```



## Quy ước

- Tất cả các câu lệnh php đều cách nhau bởi dấu “;”
- **Không phân biệt** khoảng trắng, Tab, xuống dòng trong câu lệnh

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

```
<?php  
    Print "Hello"           ;  
    print " World!";  
?>
```

- **Ghi chú** : Theo cú pháp ghi chú của C++ & Perl
  - // Đây là ghi chú
  - # Đây là ghi chú
  - /\* Đây là ghi chú nhiều dòng\*/

# Cú pháp & Quy ước trong PHP

- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- Toán tử
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng

## Khai báo biến

**\$**ten\_bien = value;

- Không khai báo kiểu dữ liệu
- Biến tự động được khởi tạo ở lần đầu tiên gán giá trị cho biến
- Tên biến :
  - Có thể bao gồm các Ký tự (A..Z, a..z), Ký số (0..9), \_, \$
  - Không được bắt đầu bằng ký số (0..9)
  - Phân biệt chữ hoa – chữ thường

Ví dụ:

\$size      \$my\_drink\_size      \$\_drinks      \$drink4you

**\$**\$2hot4u   **\$**drink-size   **x**

## Khai báo biến

### ■ Variable variables

- Cho phép thay đổi tên biến
- Ví dụ:

```
$varname = "Bien_moi";
```

```
$$varname = "xyz"; // $Bien_moi = "xyz"
```

### ■ Hằng số - Constants

- Ví dụ:

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

```
echo MY_CONST;
```

# Cú pháp & Quy ước trong PHP

- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- Toán tử
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng

## Kiểu dữ liệu

- boolean (bool)
- integer (int)
- double (float, real)
- string
- array
- object

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



## Kiểu dữ liệu (tt)

### ■ Chuyển kiểu dữ liệu

#### - Cách 1 (**automatic**)

```
$var = "100" + 15;
```

```
$var = "100" + 15.0;
```

```
$var = 39 . " Steps";
```

#### - Cách 2: (**datatype**) \$var

#### - Cách 3: **settype**(\$var, "**datatype**")

\$var	(int)\$var	(bool)\$var	(string)\$var
null	0	false	""
true	1		"1"
false	0		""
"6 feet"	6	true	
"foo"	0	true	

## Kiểu dữ liệu (tt)

- Kiểm tra kiểu dữ liệu

<b>gettype</b>	<b>is_string</b>	<b>isset</b>
<b>is_integer</b>	<b>is_array</b>	<b>unset</b>
<b>is_double</b>	<b>is_object</b>	<b>empty</b>

### Ví dụ:

```
$var = "test";  
if (isset($var))  
    echo "Variable is Set";  
if (empty($var))  
    echo "Variable is Empty";
```

## Kiểu số - int, float

- Một số hàm xử lý số

- abs	pow	decbin	rand(seed)
- ceil	sqrt	bindec	rand
- floor	log	dechex	rand(min, max)
- round	log10	hexdec	...

- Ví dụ

// Phát sinh một “mầm” ngẫu nhiên

```
$seed = (float) microtime( ) * 100000000;
```

// Khởi tạo bộ phát sinh số ngẫu nhiên

```
srand($seed);
```

// In con số ngẫu nhiên

```
print rand(); // Giá trị ngẫu nhiên từ 0 đến getmaxrand( )
```

```
print rand(1, 6); // Giá trị ngẫu nhiên từ 1 đến 6
```

## Kiểu chuỗi - string

- Toán tử nối chuỗi : dấu chấm .

```
$s = "Hello" . " World";      // $s = "Hello World"
```

- Phân biệt dấu nháy đơn và nháy kép

```
$user = "Bill";
print „Hi $user“;           // Hi $user
print "Hi $user";           // Hi Bill
print „Hi“ . $user;          // ???
print „Hi“ . „$user“;        // ???
```

- Một số hàm xử lý chuỗi

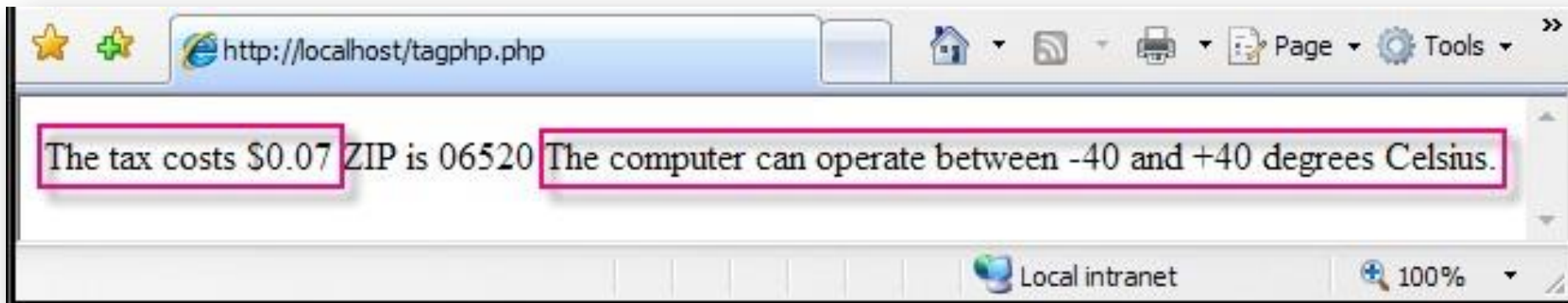
- <b>printf</b>	<b>trim</b>	<b>strtolower</b>
- <b>str_pad</b>	<b>str_replace</b>	<b>strtoupper</b>
- <b>strlen</b>	<b>substr</b>	<b>strcasecmp</b>
- ...		

## Ví dụ

```
<?
$tax = 0.075;
printf('The tax costs $%.2f', $tax);

$zip = '6520';
printf("ZIP is %05d", $zip);

$min = -40; $max = 40;
printf("The computer can operate between %+d and %+d
degrees Celsius.", $min, $max);
?>
```

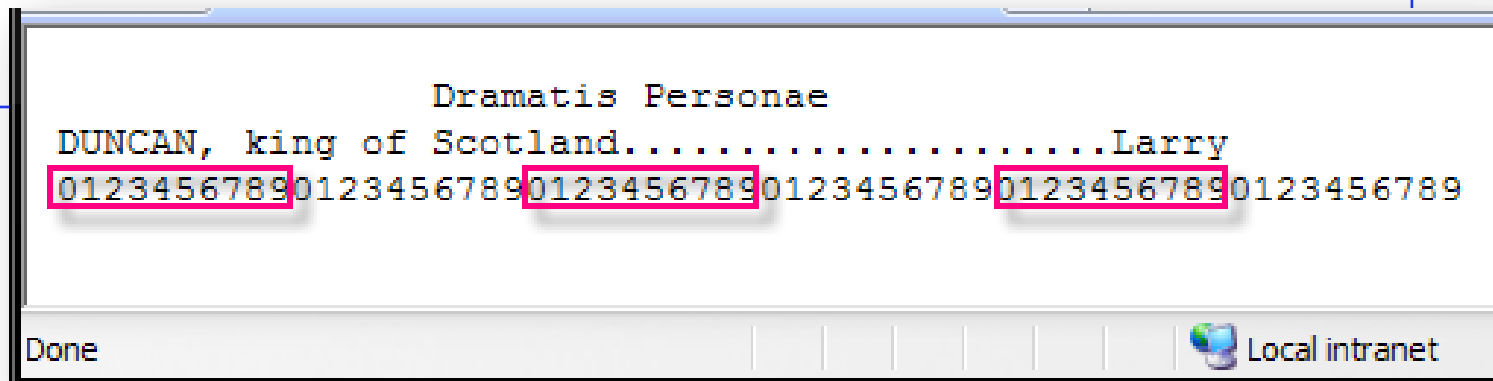
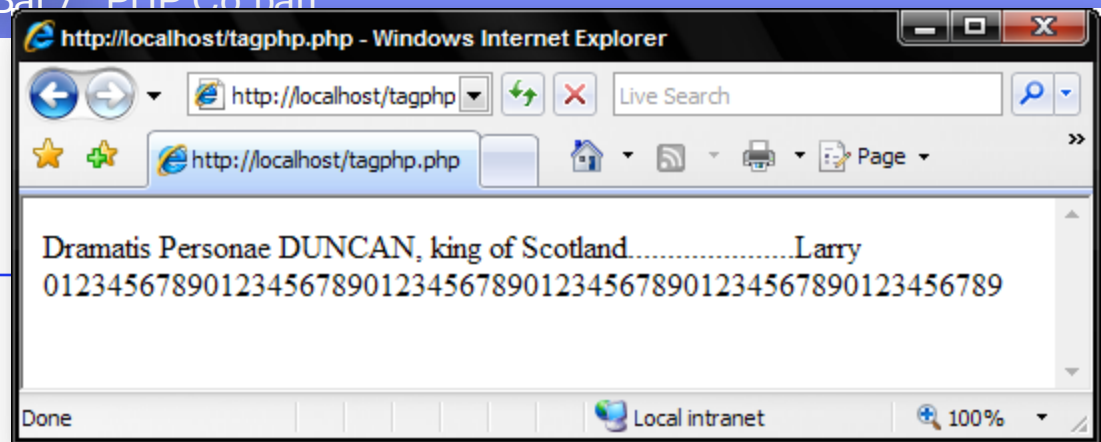


## Ví dụ

```
<?
echo "<pre>"

// Print a heading
echo str_pad("Dramatis Personae", 50, " ", STR_PAD_BOTH) . "\n";

// Print an index line
echo str_pad("DUNCAN, king of Scotland", 30, ".")
    . str_pad("Larry", 20, ".", STR_PAD_LEFT)
    . "\n";
echo "</pre>"
?>
```



# Mảng - array

## ■ Numbered array

```
$words = array("Web", "Database", "Applications");  
echo $words[0];  
$numbers = array(1=>"one", "two", "three", "four");  
echo $numbers[1];
```



## ■ Associated array

```
$array = array("first"=>1, "second"=>2, "third"=>3);  
echo $array["second"];
```



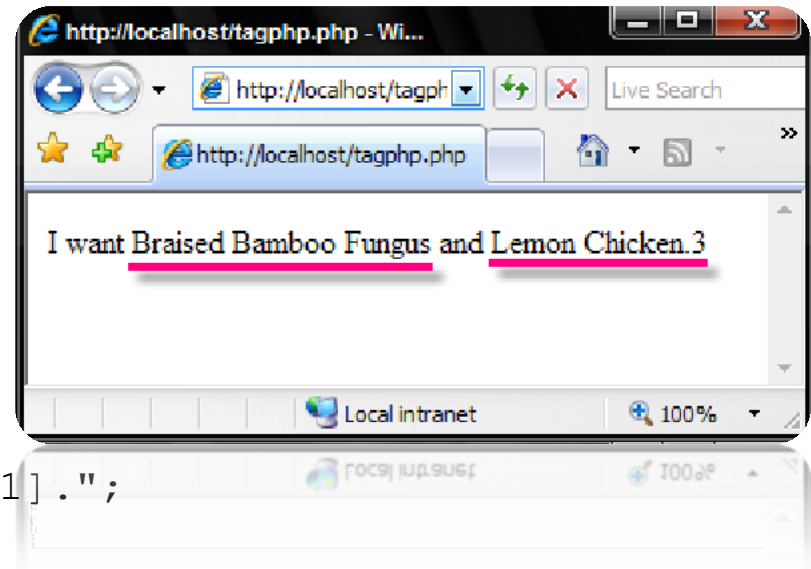
# Mảng - array

- Một số hàm xử lý trên mảng

- Count	in_array	sort	asort	ksort	usort
- min	array_reverse	rsort	arsort	krsort	uasort
- max					uksort

- Ví dụ:

```
$dinner = array(
    'Sweet Corn and Asparagus',
    'Lemon Chicken',
    'Braised Bamboo Fungus');
sort($dinner);
print "I want $dinner[0] and $dinner[1].";
$dishes = count($dinner);
print $dishes;
```





## Mảng - array

- Một số hàm liên quan đến mảng
  - `reset(array)`
  - `array_push(array, elements)` : Thêm elements vào cuối mảng
  - `array_pop(array)` : Lấy phần tử cuối ra khỏi mảng
  - `array_unshift(array, elements)` : Thêm elements vào đầu mảng
  - `array_shift(array)` : Lấy phần tử đầu ra khỏi mảng
  - `array_merge(array, array)` : kết 2 mảng lại và trả ra mảng mới
  - `shuffle(array)` : Sort random mảng
  - `sort(array, flag)` : `flag = {sort_regular, sort_numeric, sort_string, sort_locale_string}`

# Cú pháp & Quy ước trong PHP

- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- **Toán tử**
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng

# Toán tử

Loại	Toán tử	Ghi chú
	new	
	. [] ()	
Toán học	+ - * / % ++ --	
So sánh	< > <= >= != == === !==	
Luận lý	&&    ?: ,	
Xử lý bit	! ~ << >> >>> AND OR XOR	
Gán	= += -= *= /= %= >>= <<= &=  = ^= .=	
Ép kiểu	(kiểu dữ liệu)	(int) (double) (string)...

# Cú pháp & Quy ước trong PHP

- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- Toán tử
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng

## Cấu trúc điều khiển

- Điều kiện **if**
- Điều khiển **switch**
- Vòng lặp **for**
- Vòng lặp **while**
- Vòng lặp **do.. While**
- Vòng lặp **foreach**
  
- Từ khóa **break, continue**

## Điều kiện if

```
if (condition)
{
    statement[s] if true
}
else (condition)
{
    statement[s] if false
}
```

### Ví dụ:

```
$x = 5;
if ($x < 4)
    echo "$x is less than 4";
else
    print „$x isn`t less than 4“;
```

**\$x isn't less than 4**

## Điều khiển switch

```
switch (expression)
{
    case label :
        statementlist
        break;

    case label :
        statementlist
        break;

    ...

    default :
        statementlist
}
```

### Ví dụ:

```
$menu = 3;
switch ($menu) {
    case 1:
        echo "You picked one";
        break;
    case 2:
        echo "You picked two";
        break;
    case 3:
        echo "You picked three";
    case 4:
        echo "You picked four";
        break;
    default:
        echo "You picked another option";
}
```

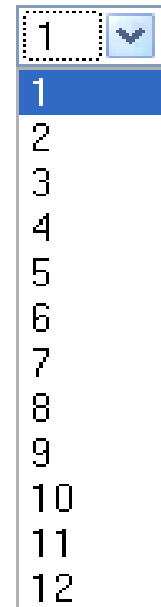
You picked three You picked four

# Vòng lặp for

```
for ([initial expression]; [condition]; [update expression])  
{  
    statement[s] inside loop  
}
```

## ■ Ví dụ:

```
print "<select>";  
for ($i = 1; $i <= 12; $i++)  
{  
    print "<option>$i</option>";  
}  
print "</select>";
```



1
2
3
4
5
6
7
8
9
10
11
12



## Vòng lặp while, do...while

### Ví dụ:

```
while (expression)
{
    statements
}
```

```
do
{
    statements
}while (expression);
```

```
$i = 1; $j = 9;

while ($i <= 10)
{
    $temp = $i * $j;
    print "$j * $i = $temp<br>";
    $i++;
}
```

9 x 1 = 9
9 x 2 = 18
9 x 3 = 27
9 x 4 = 36
9 x 5 = 45
9 x 6 = 54
9 x 7 = 63
9 x 8 = 72
9 x 9 = 81
9 x 10 = 90

# Vòng lặp foreach

```
foreach (array as variable)
{
    statements
}
```

## Ví dụ:

```
$meal = array('breakfast' => 'Walnut Bun',
              'lunch' => 'Cashew Nuts and White Mushrooms',
              'dinner' => 'Eggplant with Chili Sauce');

print "<table border=„1“>\n";
foreach ($meal as $key => $value)
{
    print "<tr><td>$key</td><td>$value</td></tr>\n";
}
print '</table>';
```

breakfast	Walnut Bun
lunch	Cashew Nuts and White Mushrooms
dinner	Eggplant with Chili Sauce

# Cú pháp & Quy ước trong PHP

- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- Toán tử
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng

# Hàm - function

```
function functionName ([parameter1]...[,parameterN])  
{  
    statement[s] ;  
}
```

```
function functionName ([parameter1]...[,parameterN])  
{  
    statement[s] ;  
    return .... ;  
}
```

## Hàm – Phạm vi biến

```
<?php
function doublevalue($var=5)
{
    global $temp;
    $temp = $var * 2;
}
```

```
$temp = 5;
doublevalue();
echo "\$temp is: $temp";
?>
```

**\$temp = 5**

**\$temp = 10**

## Hàm – Tham trị vs Tham biến

```
<?php
function doublevalue( & $var)
{
    $var = $var * 2;
}

$variable = 5;
doublevalue($variable);
echo "\$variable is: $variable";
?>
```

**\$variable = 5**

**\$variable =10**

## Hàm – include & require

```
// functions.inc
<?php
function bold($string)
{
    echo "<b>" . $string . "</b>\n";
}
?>
```

```
// index.php
<html>
<head>
    <title>Simple Function Call</title>
</head>
<body bgcolor="#ffffff">
    <?
require "functions.inc";

    bold("this is bold");

    $myString = "this is bold";
    bold($myString);
    ?>
</body></html>
```

# Cú pháp & Quy ước trong PHP

- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- Toán tử
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng



# Lớp đối tượng - class

```
class class_name() [extends superclass_name]
{
    var $attribute;
    ...
    function method_name()
    {
        $this->attribute = ...;
    }
    ...
}

$a = new class_name (...);
```

## Lớp đối tượng - class

- Ví dụ:

```
class Counter {  
    var $count = 0;  
    var $startPoint = 0;  
  
    function increment( ) {  
        $this->count++;  
    }  
}  
  
$aCounter = new Counter;  
$aCounter->increment( );  
  
echo $aCounter->count; // prints 1  
$aCounter->count = 101;
```