



Chương 6 LẬP TRÌNH WEB VỚI PHP

DHTML

JAVA
SCRIPT

LẬP TRÌNH
WEB

PHP

Trình bày: Nguyễn Phú Trường

- ❖ Giới thiệu về PHP
- ❖ Biến, kiểu dữ liệu, phép toán
- ❖ Lệnh điều khiển
- ❖ Hàm
- ❖ PHP kết hợp với forms
- ❖ Cookies, SSI (Server side includes), Date
- ❖ PHP-MySQL

- ❖ PHP viết tắt **PHP: Hypertext Preprocessor**
- ❖ Ngôn ngữ script chạy trên server
- ❖ PHP hỗ trợ để làm việc với nhiều hệ quản trị cơ sở dữ liệu khác nhau:
 - MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC,...
- ❖ Phần mềm mã nguồn mở, miễn phí
- ❖ Có thể tải về và sử dụng

- ❖ PHP scripts chứa văn bản, thẻ HTML, script
- ❖ Sử dụng phần mở rộng tên file : .php, .php3, .phtml
- ❖ PHP scripts sẽ trả về kết quả cho trình duyệt một plain HTML

MYSQL LÀ GÌ ?



- ❖ Hệ quản trị cơ sở dữ liệu
- ❖ Dùng cho các ứng dụng vừa và nhỏ
- ❖ Hỗ trợ SQL chuẩn
- ❖ Phần mềm mã nguồn mở, miễn phí
- ❖ Chạy trên nhiều platforms (Unix, Linux, Windows)
- ❖ Phổ biến

- ❖ Web động chạy trên nhiều platforms khác nhau
- ❖ PHP được kết hợp với MySQL là cross-platform

TẠI SAO DÙNG PHP?



- ❖ Chạy trên nhiều nền khác nhau (Windows, Linux, Unix,..)
- ❖ Phần mềm mã nguồn mở, miễn phí, có thể tải về từ www.php.net
- ❖ Tương thích với hầu hết các web server (Apache, IIS,..)
- ❖ Dễ học và phát triển nhanh các ứng dụng trên Web

- ❖ Cài web server (Apache, IIS, ..):
 - www.apache.org
- ❖ Cài MySQL:
 - www.mysql.com
- ❖ Cài PHP:
 - www.php.net
- ❖ Cài Vertrigo hay Xamp hay EasyPHP,.. chứa cả Apache, MySQL, PHP

- ❖ Giới thiệu về PHP
- ❖ Biến, kiểu dữ liệu, phép toán
- ❖ Lệnh điều khiển
- ❖ Hàm
- ❖ PHP kết hợp với forms
- ❖ Cookies, SSI (Server side includes), Date
- ❖ PHP-MySQL

❖ Cú pháp

- PHP scripts chứa text, thẻ HTML, script
- Ví dụ : in ra màn hình chuỗi “Hello World”

```
<html>
```

```
<body>
```

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

```
</body>
```

```
</html>
```

- ❖ Khởi lệnh PHP script bắt đầu với `<?php` và kết thúc bởi `?>`
- ❖ Khởi lệnh có thể được đặt bất cứ đâu trong tài liệu
- ❖ Mỗi lệnh cách nhau bởi dấu `;`
- ❖ Có 2 lệnh cơ bản để in văn bản ra màn hình: `echo` và `print`
- ❖ Chú thích trong chương trình
 - `//` chú thích là 1 dòng đơn
 - `/*` chú thích là 1 đoạn văn bản `*/`

❖ Ví dụ :

```
<?php
```

```
echo "This is a test"; // This is a one-line c++ style  
comment
```

```
/* This is a multi line comment  
yet another line of comment */
```

```
echo("This is yet another test");  
print "Hello World";  
print("Hello World");
```

```
?>
```

- ❖ Được dùng để chứa giá trị hay biểu thức
- ❖ Qui cách
 - Biến được bắt đầu bởi dấu \$ theo sau tên
 - Tên biến bắt đầu bằng 1 ký tự chữ hoặc _
 - Tên biến chỉ có thể chứa các ký tự chữ (A-z), số (0-9) và dấu _
 - Phân biệt giữa ký tự thường và hoa
 - Tên biến không chứa khoảng trắng

- ❖ Biến được tạo ra tại thời điểm gán giá trị
- ❖ Gán giá trị với =
- ❖ Thí dụ:
 - `$txt="Hello world!";`
`$x=5;`
- ❖ Không phải xác định kiểu khi khai báo biến
- ❖ PHP tự chuyển đổi biến tới kiểu phù hợp phù thuộc vào giá trị của biến

❖ Ví dụ:

```
<?php
```

```
$var = 'Bob';
```

```
$Var = 'Joe';
```

```
echo "$var, $Var"; // xuất ra "Bob, Joe"
```

```
$4site = 'not yet'; // không hợp lệ; bắt đầu 1 số
```

```
$_4site = 'not yet'; // hợp lệ; bắt đầu với _
```

```
$täyte = 'mansikka'; // hợp lệ; 'ä' có mã ASCII mở rộng 228.
```

```
?>
```

❖ Ví dụ:

```
<?php
$foo = 'Bob';           // Gán giá trị 'Bob' tới $foo
$bar = &$foo;           // Tham chiếu $foo bằng $bar.
$bar = "My name is $bar"; // Thay đổi $bar...
echo $bar;              // My name is Bob
echo $foo;              // My name is Bob
?>
```


❖ Ví dụ:

```
<?php
$foo = 'Bob';
echo $foo;           // Bob
$foo = 12
echo $foo;           // 12
$foo = array(1, 2, 3, 4, 5);
for($i = 0; $i < 5; $i++)
    echo $foo[$i] . "<br>";
?>
```

- ❖ `$GLOBALS`: tất cả biến có phạm vi toàn cục của script
- ❖ `$_SERVER`: tập hợp biến môi trường của Web server
- ❖ `$_GET`, `$_POST`: biến được cung cấp các chuỗi query URL cho script
- ❖ `$_COOKIE`: biến cung cấp HTTP_cookies cho script
- ❖ `$_FILES`: biến cung cấp HTTP POST file uploads cho script
- ❖ `$_ENV`: biến cung cấp môi trường cho script
- ❖ `$_REQUEST`: cung cấp `$_GET`, `$_POST`, `$_COOKIE`

- ❖ Phạm vi biến là một phần của script nơi biến có thể được tham chiếu.
- ❖ PHP có 4 phạm vi biến khác nhau:
 - Cục bộ (local)
 - Toàn cục (global)
 - Tĩnh (static)
 - Tham số (parameter)

❖ Biến được khai báo bên trong một hàm là cục bộ và chỉ có thể được truy xuất bên trong hàm đó.

❖ Thí dụ:

- `<?php`
`$x=5; // global scope`

```
function myTest()  
{  
    echo $x; // local scope  
}
```

```
myTest();  
?>
```

❖ Cục bộ

❖ Thí dụ :

```
<?php
```

```
$a = 1; /* phạm vi biến toàn cục */
```

```
function Test() {
```

```
    $a = 10;
```

```
    echo " in Test a = " . $a; /* tham chiếu tới biến phạm vi cục bộ */
```

```
}
```

```
Test();
```

```
echo "<br> out Test a = " . $a;
```

```
?>
```

- ❖ Biến được định nghĩa bên ngoài hàm, có phạm vi toàn cục.
- ❖ Các biến toàn cục có thể được truy xuất từ bất cứ phần nào của script, ngoại trừ từ bên trong một hàm.
- ❖ Để truy xuất biến toàn cục từ trong một hàm, dùng từ khóa **global**.
- ❖ PHP cũng lưu tất cả các biến toàn cục trong một mảng được gọi là `$GLOBAL[index]`.
 - Index giữ tên của biến
 - Mảng này cũng được truy xuất bên trong các hàm và có thể được dùng để cập nhật biến toàn cục trực tiếp.

❖ Ví dụ :

```
<?php
```

```
$a = 1; //phạm vi toàn cục
```

```
$b = 2; //phạm vi toàn cục
```

```
function Sum() {
```

```
    global $a, $b;
```

```
    $b = $a + $b;
```

```
}
```

```
Sum();
```

```
echo $b;
```

```
?>
```

❖ Ví dụ :

```
<?php
```

```
$a = 1;
```

```
$b = 2;
```

```
function Sum() {
```

```
    $GLOBALS['b'] = $GLOBALS['a'] + $GLOBALS['b'];
```

```
}
```

```
Sum();
```

```
echo $b;
```

```
?>
```


- ❖ Khi một hàm hoàn thành, tất cả biến của nó thường được xóa.
- ❖ Tuy nhiên, đôi khi chúng ta muốn biến cục bộ không bị xóa
- ❖ Để làm điều này, dùng từ khóa **static** khi chúng ta khai báo biến
- ❖ Sau mỗi lần hàm được gọi, biến sẽ vẫn có thông tin nó chứa từ lần cuối cùng hàm được gọi.
- ❖ Chú ý:
 - Biến vẫn là cục bộ đối với hàm

❖ Ví dụ :

```
<?php
function Test() {
    static $a = 10;
    echo " in Test a = " . $a;
    $a++;
}
Test(); // 10
Test(); // 11
?>
```

- ❖ Một tham số là một biến cục bộ mà giá trị được truyền tới hàm bởi mã gọi
- ❖ Các tham số được khai báo trong một danh sách tham số của khai báo hàm.
- ❖ Ví dụ:
 - ```
<?php
function myTest($x)
{
 echo $x;
}
myTest(5);

?>
```

- ❖ Giới thiệu về PHP
- ❖ Biến, kiểu dữ liệu, phép toán
- ❖ Lệnh điều khiển
- ❖ Hàm
- ❖ PHP kết hợp với forms
- ❖ Cookies, SSI (Server side includes), Date
- ❖ PHP-MySQL

## ❖ Kiểu dữ liệu cơ bản

- Số nguyên : 4 bytes, số có dấu
- Số thực
- Luận lý : TRUE/FALSE
- Chuỗi ký tự

## ❖ Kiểu dữ liệu phức hợp

- Mảng
- Đối tượng
- Kiểu giả
- ..

## ❖ Thí dụ : số nguyên, số thực

```
<?php
```

```
$a = 1234; // decimal number
```

```
$a = -123; // a negative number
```

```
$a = 0123; // octal number (equivalent to 83 decimal)
```

```
$a = 0x1A; // hexadecimal number (equivalent to 26 decimal)
```

```
$b = 1.234;
```

```
$c = 1.2e3;
```

```
$d = 7E-10;
```

```
?>
```

## ❖ Luận lý

```
<?php
$foo = TRUE; // assign the value TRUE to $foo
if ($action == "show_version") {
 echo "The version is 1.23";
}
// this is not necessary...
if ($show_separators == TRUE) {
 echo "<hr>\n";
}
// ...because you can simply type
if ($show_separators) {
 echo "<hr>\n";
} ?>
```

```
<?php
$beer = 'Heineken';
echo "$beer's taste is great"; // works, "" is an invalid character for
 varnames
echo "He drank some $beers"; // won't work, 's' is a valid character for
 varnames
echo "He drank some ${beer}s"; // works
echo "He drank some {$beer}s"; // works
$str = 'This is a test.';
$third = $str{2}; // Get the third character of a string
$str = "This is still a test.";
$last = $str{strlen($str)-1}; // Get the last character of a string.
$str = 'Look at the sea';
$str{strlen($str)-1} = 'e'; // Modify the last character of a string
```



- ❖ Lưu nhiều giá trị trong một biến đơn
- ❖ Truy xuất các giá trị bằng cách tham chiếu tới số chỉ mục
- ❖ Thí dụ:
  - ```
<?php  
$cars=array("Volvo","BMW","Toyota");  
echo "I like " . $cars[0] . ", " . $cars[1] . " and " . $cars[2] .  
" . ";  
?>
```

- ❖ Trong PHP hàm `array()` được dùng để tạo một mảng
- ❖ Trong PHP, có 3 loại mảng:
 - Mảng được lập chỉ mục -Mảng với chỉ mục số
 - Mảng kết hợp-mảng với khóa được đặt tên
 - Mảng đa chiều-mảng chứa một hay nhiều mảng

❖ Có hai cách để tạo mảng được lập chỉ mục:

- Chỉ mục có thể được gán tự động (luôn bắt đầu từ 0).

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

- Hoặc chỉ mục được gán thủ công:

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

❖ Thí dụ:

```
<?php  
$cars=array("Volvo","BMW","Toyota");  
echo "I like " . $cars[0] . ", " . $cars[1] . " and " . $cars[2] . ".";  
?>
```

❖ Lấy chiều dài của mảng dùng hàm count()

- <?php
\$cars=array("Volvo","BMW","Toyota");
echo count(\$cars);
?>

❖ Lập quan một mảng được lập chỉ mục

- <?php
\$cars=array("Volvo","BMW","Toyota");
\$arlength=count(\$cars);
for(\$x=0;\$x<\$arlength;\$x++)
{
echo \$cars[\$x];
echo "
";
}?
>

- ❖ Mảng kết hợp là mảng dùng các khóa được đặt tên để gán cho chúng.
- ❖ Có hai cách để tạo một mảng kết hợp:
 - `$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");`
 - Hay:
`$age['Peter']="35";`
`$age['Ben']="37";`
`$age['Joe']="43";`
- ❖ Sau đó các khóa được đặt tên được dùng trong script:
 - `<?php`
`$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");`
`echo "Peter is " . $age['Peter'] . " years old.";`
`?>`

❖ Lặp qua mảng kết hợp:

- <?php
\$age=array("Peter"=>"35","Ben"=>"37","Joe"=>"43");

foreach(\$age as \$x=>\$x_value)

```
{  
    echo "Key=" . $x . ", Value=" . $x_value;  
    echo "<br>";  
}  
?>
```

- ❖ Mảng đa chiều là một mảng chứa một hay nhiều mảng.
- ❖ Trong mảng đa chiều, mỗi phần tử trong mảng chính cũng có thể là một mảng.
- ❖ Mỗi phần tử trong mảng con lại có thể chứa mảng khác.
- ❖ Ví dụ 1:

```
<?php
// A two-dimensional array:
$scars = array
(
    array("Volvo",100,96),
    array("BMW",60,59),
    array("Toyota",110,100)
);
```

❖ Thí dụ 2:

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

❖ Thí dụ 3:

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


PHÉP TOÁN SỐ HỌC



| Phép toán | Tên | Thí dụ | Kết quả |
|-----------|------------------|------------------------------------|-------------|
| $x + y$ | Cộng | $2 + 2$ | 4 |
| $x - y$ | Trừ | $5 - 2$ | 3 |
| $x * y$ | Nhân | $5 * 2$ | 10 |
| x / y | Chia | $15 / 5$ | 3 |
| $x \% y$ | Chia lấy phần dư | $5 \% 2$
$10 \% 8$
$10 \% 2$ | 1
2
0 |
| $- x$ | Âm | $- 2$ | |
| $a . b$ | Cộng chuỗi | "Hi" . "Ha" | HiHa |

| Gán | Tương đương | Mô tả |
|------------|--------------|--|
| $x = y$ | $x = y$ | Toán hạng bên trái được gán giá trị của biểu thức bên phải |
| $x += y$ | $x = x + y$ | Cộng |
| $x -= y$ | $x = x - y$ | Trừ |
| $x *= y$ | $x = x * y$ | Nhân |
| $x /= y$ | $x = x / y$ | Chia |
| $x \% = y$ | $x = x \% y$ | Chia lấy phần dư |
| $a .= b$ | $a = a . b$ | Nối hai chuỗi |

PHÉP TOÁN TĂNG/GIẢM



| Toán tử | Tên | Mô tả |
|-------------------|------------|---------------------------------|
| <code>++ x</code> | Tăng trước | Tăng x lên 1, sau đó trả về x |
| <code>x ++</code> | Tăng sau | Trả về x, sau đó tăng x lên 1 |
| <code>-- x</code> | Giảm trước | Giảm x xuống 1, sau đó trả về x |
| <code>x --</code> | Giảm sau | Trả về x, sau đó giảm x xuống 1 |

PHÉP TOÁN SO SÁNH



| Toán tử | Tên | Mô tả | Thí dụ |
|-----------|------------------|--|-------------------------------|
| $x == y$ | Bằng | True nếu x là bằng y | $5==8$ trả về false |
| $x === y$ | Đồng nhất | True nếu x là bằng y và chúng cùng kiểu | $5===\text{"5"}$ trả về false |
| $x != y$ | Không bằng | True nếu x là không bằng y | $5!=8$ trả về true |
| $x <> y$ | Không bằng | True nếu x là không bằng y | $5<>8$ trả về true |
| $x !== y$ | Không đồng nhất | True nếu x là không bằng y hay chúng khác kiểu | $5!==\text{"5"}$ trả về true |
| $x > y$ | Lớn hơn | True nếu x lớn hơn y | $5>8$ trả về false |
| $x < y$ | Nhỏ hơn | True nếu x nhỏ hơn y | $5<8$ trả về true |
| $x >= y$ | Lớn hơn hay bằng | True nếu x lớn hơn hay bằng y | $5>=8$ trả về false |
| $x <= y$ | Nhỏ hơn hay bằng | True nếu x nhỏ hơn hay bằng y | $5<=8$ trả về true |

PHÉP TOÁN LUẬN LÝ



| Toán tử | Tên | Mô tả | Thí dụ |
|---------|-----|---|---|
| x and y | And | True nếu cả hai là đúng | x=6
y=3
(x < 10 and y > 1)
trả về true |
| x or y | Or | True nếu một trong hai là true | x=6
y=3
(x==6 or y==5) trả về true |
| x xor y | Xor | True nếu một trong hai là true nhưng không cả hai | x=6
y=3
(x==6 xor y==3) trả về false |

PHÉP TOÁN LUẬN LÝ



| Toán tử | Tên | Mô tả | Thí dụ |
|----------------|-----|--|--|
| $x \ \&\& \ y$ | And | True cả x và y đều là true | $x=6$
$y=3$
$(x < 10 \ \&\& \ y > 1)$
trả về true |
| $x \ \ y$ | Or | True nếu ít nhất một trong hai x và y là đạt | $x=6$
$y=3$
$(x==5 \ \ y==5)$ trả về false |
| $! \ x$ | Not | True nếu z không là true | $x=6$
$y=3$
$!(x==y)$ trả về true |

- ❖ Giới thiệu về PHP
- ❖ Biến, kiểu dữ liệu, phép toán
- ❖ Lệnh điều khiển
- ❖ Hàm
- ❖ PHP kết hợp với forms
- ❖ Cookies, SSI (Server side includes), Date
- ❖ PHP-MySQL



if

■ Cú pháp :

if (condition)

code to be executed if condition is true;

else

code to be executed if condition is false;

■ Thí dụ :

<?php

\$d=date("D");

if (\$d=="Fri")

echo "Have a nice weekend!";

else

echo "Have a nice day!";

❖ Switch

■ Cú pháp :

```
switch (expression) {
```

```
case label1:
```

```
    code to be executed if expression = label1;
```

```
    break;
```

```
case label2:
```

```
    code to be executed if expression = label2;
```

```
    break;
```

```
default:
```

```
    code to be executed
```

```
    if expression is different
```

```
    from both label1 and label2;
```

❖ switch

■ Ví dụ :

<?php

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";

}

❖ while

■ Cú pháp :

while (condition)

code to be executed;

■ Ví dụ :

```
<?php
```

```
$i=1;
```

```
while($i<=5) {
```

```
    echo "The number is " . $i . "<br />";
```

```
    $i++;
```

```
}
```

```
?>
```

❖ Do ... while

■ Cú pháp :

```
do {  
    code to be executed;  
} while (condition);
```

■ Ví dụ :

```
<?php  
$i=0;  
do {  
    $i++;  
    echo "The number is " . $i . "<br />";  
} while ($i<5);  
?>
```

❖ for

■ Cú pháp :

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

■ Ví dụ :

```
<?php  
for ($i=1; $i<=5; $i++)  
{  
    echo "Hello World!<br />";  
}  
?>
```

❖ foreach

■ Cú pháp :

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

■ Ví dụ :

```
<?php  
$arr=array("one", "two", "three");  
foreach ($arr as $value)  
{  
    echo "Value: " . $value . "<br />";  
}  
?>
```

- ❖ Giới thiệu về PHP
- ❖ Biến, kiểu dữ liệu, phép toán
- ❖ Lệnh điều khiển
- ❖ Hàm
- ❖ PHP kết hợp với forms
- ❖ Cookies, SSI (Server side includes), Date
- ❖ PHP-MySQL

- Function Reference
- Apache-specific Function:
- Advanced PHP debugger
- Array Functions
- Aspell functions [deprecat
- BCMath Arbitrary Precisi
- PHP bytecode Compiler
- Bzip2 Compression Funct
- Calendar Functions
- CCVS API Functions [dep
- Classkit Functions
- Class/Object Functions
- COM and .Net (VWindows)
- ClibPDF Functions
- Crack Functions
- Character Type Functions
- CURL, Client URL Library
- Cybercash Payment Func
- Cyrus IMAP administratio
- Date and Time Functions
- Database (dbm-style) Abs
- dBase Functions
- DBM Functions [deprecat
- DB++ Functions
- dbx Functions
- Direct IO Functions
- Directory Functions
- DOM Functions
- DOM XML Functions
- .NET Functions
- Error Handling and Loggin
- Program Execution Funct
- Exif Functions

?????

?????

VI. Function Reference

????

- I. [.NET Functions](#)
- II. [Apache-specific Functions](#)
- III. [Alternative PHP Cache](#)
- IV. [Advanced PHP debugger](#)
- V. [Array Functions](#)
- VI. [Aspell functions \[deprecated\]](#)
- VII. [BCMath Arbitrary Precision Mathematics Functions](#)
- VIII. [PHP bytecode Compiler](#)
- IX. [Bzip2 Compression Functions](#)
- X. [Calendar Functions](#)
- XI. [CCVS API Functions \[deprecated\]](#)
- XII. [Class/Object Functions](#)
- XIII. [Classkit Functions](#)
- XIV. [ClibPDF Functions](#)
- XV. [COM and .Net \(Windows\)](#)
- XVI. [Crack Functions](#)
- XVII. [Character Type Functions](#)
- XVIII. [CURL, Client URL Library Functions](#)
- XIX. [Cybercash Payment Functions](#)
- XX. [Credit Mutuel CyberMUT functions](#)
- XXI. [Cyrus IMAP administration Functions](#)
- XXII. [Date and Time Functions](#)
- XXIII. [DB++ Functions](#)
- XXIV. [Database \(dbm-style\) Abstraction Layer Functions](#)
- XXV. [dBase Functions](#)
- XXVI. [DBM Functions \[deprecated\]](#)
- XXVII. [dbx Functions](#)
- XXVIII. [Direct IO Functions](#)
- XXIX. [Directory Functions](#)
- XXX. [DOM Functions](#)
- XXXI. [DOM XML Functions](#)
- XXXII. [Error Handling and Logging Functions](#)
- XXXIII. [Exif Functions](#)

❖ Hàm

■ Cú pháp :

<?php

function foo(\$arg_1, \$arg_2, /* ..., */ \$arg_n)

{

echo "Example function.\n";

return \$retval;

}

?>

17-2. Conditional functions

```
<?php

$makefoo = true;

/* We can't call foo() from here
   since it doesn't exist yet,
   but we can call bar() */

bar();

if ($makefoo) {
    function foo()
    {
        echo "I don't exist until program execution reaches me.\n";
    }
}

/* Now we can safely call foo()
   since $makefoo evaluated to true */

if ($makefoo) foo();

function bar()
{
    echo "I exist immediately upon program start.\n";
}

?>
```

17-3. Functions within functions

```
<?php
function foo()
{
    function bar()
    {
        echo "I don't exist until foo() is called.\n";
    }
}

/* We can't call bar() yet
   since it doesn't exist. */

foo();

/* Now we can call bar(),
   foo()'s processing has
   made it accessible. */

bar();

?>
```

17-4. Recursive functions

```
<?php
function recursion($a)
{
    if ($a < 20) {
        echo "$a\n";
        recursion($a + 1);
    }
}

?>
```

❖ Tham số

- Truyền tham số : giá trị, tham chiếu
- Hàm : `func_num_args()`, `func_get_arg()`
- Ví dụ tham số là mảng:

```
<?php  
function takes_array($input) {  
    echo "$input[0] + $input[1] = ", $input[0]+$input[1];  
}  
?>
```

❖ Tham số

- Ví dụ tham số có giá trị mặc định :

```
<?php  
function makecoffee($type = "cappuccino")  
{  
    return "Making a cup of $type.<br>";  
}  
echo makecoffee();  
echo makecoffee("espresso");  
?>
```

❖ Tham số

- Ví dụ truyền tham chiếu :

```
<?php
```

```
function add_some_extra(&$string)
```

```
{
```

```
    $string .= 'and something extra.';
```

```
}
```

```
$str = 'This is a string, ';
```

```
add_some_extra($str);
```

```
echo $str; // outputs 'This is a string, and something extra.'
```

```
?>
```

❖ Giá trị trả về

■ Ví dụ :

```
<?php  
function square($num)  
{  
    return $num * $num;  
}  
echo square(4); // outputs '16'.  
?>
```

❖ Giá trị trả về

▪ Ví dụ :

```
<?php  
function small_numbers()  
{  
    return array (0, 1, 2);  
}  
list ($zero, $one, $two) = small_numbers();  
?>
```


❖ Giá trị trả về

▪ Ví dụ :

```
<?php  
function &returns_reference()  
{  
    return $someref;  
}  
$newref =& returns_reference();  
?>
```

- ❖ Giới thiệu về PHP
- ❖ Biến, kiểu dữ liệu, phép toán
- ❖ Lệnh điều khiển
- ❖ Hàm
- ❖ PHP kết hợp với forms
- ❖ Cookies, SSI (Server side includes), Date
- ❖ PHP-MySQL

❖ PHP kết hợp với HTML Form

- Hầu hết các thành phần của HTML Form đều được sử dụng trong các PHP script
- Sử dụng biến `$_GET` hay `$_POST` để truy xuất đến các thành phần của HTML Form
- Ví dụ : trang web là `welcome.html` nội dung như sau

```
<html>
```

```
<body>
```

```
<form action="welcome.php" method="POST">
```

```
Enter your name: <input type="text" name="name">
```

```
Enter your age: <input type="text" name="age">
```

```
<input type="submit" value="welcome">
```

```
</form>
```

```
</body>
```

```
</html>
```

❖ PHP kết hợp với HTML Form

- PHP script "**welcome.php**" sử dụng biến `$_POST` để truy xuất đến các thành phần của HTML Form do sử dụng **method="POST"**
- PHP script `welcome.php` nội dung như sau

```
<html>
```

```
<body>
```

```
Welcome <?php echo $_POST["name"]; ?>.<br>
```

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

```
</body>
```

```
</html>
```

- ❖ Giới thiệu về PHP
- ❖ Biến, kiểu dữ liệu, phép toán
- ❖ Lệnh điều khiển
- ❖ Hàm
- ❖ PHP kết hợp với forms
- ❖ Cookies, SSI (Server side includes), Date
- ❖ PHP-MySQL

❖ Cookie

- Thường được sử dụng để xác định một user
- Server ghi 1 tập tin cookie lên web client
- PHP cho phép tạo và đọc lại những giá trị từ cookie
- Hàm tạo cookie : `setcookie(name, value, expire, path, domain)`
- Được đặt trước thẻ `<html>`
- Ví dụ :

```
<?php setcookie("uname", $name, time()+36000); ?>
```

```
<html>
```

```
<body>
```

```
<p> A cookie was set on this page! The cookie will be active when the client has sent the  
cookie back to the server. </p>
```

```
</body>
```

```
</html>
```

❖ Cookie

- Hàm `isset()` để đọc lại cookie đã được tạo

- Ví dụ :

```
<html>
```

```
<body>
```

```
<?php
```

```
if (isset($_COOKIE["uname"]))
```

```
    echo "Welcome " . $_COOKIE["uname"] . " !<br />";
```

```
else
```

```
    echo "You are not logged in!<br />";
```

```
?>
```

```
</body>
```

```
</html>
```

❖ SSI

- Chèn đoạn code chương trình của một file vào file khác trước khi thực thi
- Sử dụng hàm require()
- Ví dụ :

```
<html>
```

```
<body>
```

```
<?php require("header.htm"); ?>
```

```
<p> Some text </p> <p>Some text</p>
```

```
</body>
```

```
</html>
```


| Cú pháp : string date (date_format[,int timestamp])

Date Formats

The table below shows the characters that may be used in the format string:

Character	Description
a	"am" or "pm"
A	"AM" or "PM"
B	Swatch Internet time (000-999)
d	Day of the month with a leading zero (01-31)
D	Three characters that represents the day of the week (Mon-Sun)
F	The full name of the month (January-December)
g	The hour in 12-hour format without a leading zero (1-12)
G	The hour in 24-hour format without a leading zero (0-23)
h	The hour in 12-hour format with a leading zero (01-12)
H	The hour in 24-hour format with a leading zero (00-23)
i	The minutes with a leading zero (00-59)
I	"1" if the date is in daylight savings time, otherwise "0"
j	Day of the month without a leading zero (1-31)
l	The full name of the day (Monday-Sunday)

HÀM THỜI GIAN



L	"1" if the year is a leap year, otherwise "0"
m	The month as a number, with a leading zero (01-12)
M	Three letters that represents the name of the month (Jan-Dec)
n	The month as a number without a leading zero (1-12)
O	The difference to Greenwich time (GMT) in hours
r	An RFC 822 formatted date (e.g. "Tue, 10 Apr 2005 18:34:07 +0300")
s	The seconds with a leading zero (00-59)
S	The English ordinal suffix for the day of the month (st, nd, rd or th)
t	The number of days in the given month (28-31)
T	The local time zone (e.g. "GMT")
U	The number of seconds since the Unix Epoch (January 1 1970 00:00:00 GMT)
w	The day of the week as a number (0-6, 0=Sunday)
W	ISO-8601 week number of year, weeks starting on Monday
Y	The year as a 4-digit number (e.g. 2003)
y	The year as a 2-digit number (e.g. 03)
z	The day of the year as a number (0-366)

n Date()

| Ví dụ :

```
<?php
```

```
//Prints something like: Monday
```

```
echo date("l");
```

```
//Prints something like: Monday 15th of January 2003 05:51:38 AM
```

```
echo date("l dS of F Y h:i:s A");
```

```
//Prints something like: Monday the 15th
```

```
echo date("l \\t\\h\\e jS");
```

```
?>
```

- ❖ Giới thiệu về PHP
- ❖ Biến, kiểu dữ liệu, phép toán
- ❖ Lệnh điều khiển
- ❖ Hàm
- ❖ PHP kết hợp với forms
- ❖ Cookies, SSI (Server side includes), Date
- ❖ PHP-MySQL

n MySQL

- | Download : www.mysql.com, cài đặt
- | Có thể cài thêm giao diện quản trị
- | Hoặc sử dụng trình mysql (client)

```
mysql -u root -p
```

```
Enter password: *****
```

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
```

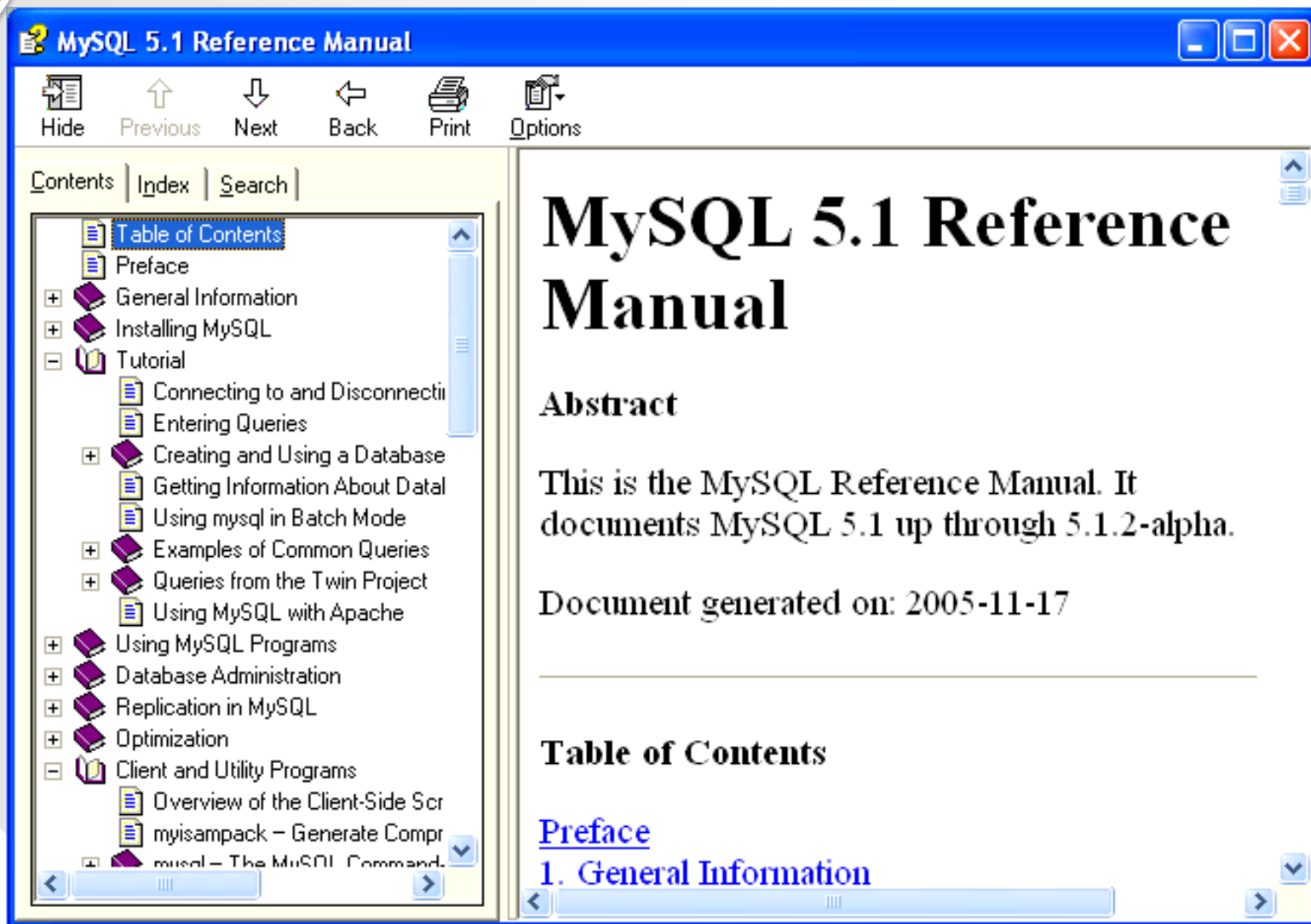
```
Your MySQL connection id is 4 to server version: 5.0.15-nt
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
```

```
mysql>
```

n Lệnh cơ bản MySQL

- | Tạo xóa cơ sở dữ liệu : `create (drop) database dbname`
- | Tạo xóa người dùng : `create (drop) user uname`
- | Tạo xóa quyền truy cập : `grant (revoke) ...`
- | Tạo xóa bảng : `create (drop) table tname`
- | Chèn mẫu tin : `insert into tname values (...)`
- | Xóa mẫu tin : `delete ... from tname where ...`
- | Cập nhật : `update tname set colname = value ...`



MySQL Administrator - root@127.0.0.1:3306

File Edit View Tools Window Help

Server Information
Service Control
Startup Variables
User Administration
Server Connections
Health
Server Logs
Replication Status
Backup
Restore
Catalogs

Schemata

- information_schema
- mydb
- mysql
- test

Schema Tables Schema Indices Views Stored procedures

mysql
All mysql indices

Index Name	Table Name	Type	Unique	Not Null
user_info_Full_name	user_info	BTREE		
User	db	BTREE		NOT NULL
PRIMARY	user_info	BTREE	UNIQUE	NOT NULL
PRIMARY	user	BTREE	UNIQUE	NOT NULL
PRIMARY	time_zone_transition_type	BTREE	UNIQUE	NOT NULL
PRIMARY	time_zone_transition	BTREE	UNIQUE	NOT NULL
PRIMARY	time_zone_name	BTREE	UNIQUE	NOT NULL
PRIMARY	time_zone_leap_second	BTREE	UNIQUE	NOT NULL
PRIMARY	time_zone	BTREE	UNIQUE	NOT NULL
PRIMARY	tables_priv	BTREE	UNIQUE	NOT NULL
PRIMARY	procs_priv	BTREE	UNIQUE	NOT NULL
PRIMARY	proc	BTREE	UNIQUE	NOT NULL
PRIMARY	host	BTREE	UNIQUE	NOT NULL
PRIMARY	help_topic	BTREE	UNIQUE	NOT NULL
PRIMARY	help_relation	BTREE	UNIQUE	NOT NULL
PRIMARY	help_keyword	BTREE	UNIQUE	NOT NULL

Num. of Indices: 25 Num. of Columns: 45 Num. of Unique: 21 Num. of NotNull: 24

Refresh

n Ví dụ :

| Tạo cơ sở dữ liệu mydb : `create database mydb;`

| Tạo bảng Person

```
mysql> use mydb;
```

Database changed

```
mysql> CREATE TABLE Person
```

```
-> (
```

```
-> lastname varchar(30),
```

```
-> firstname varchar(10),
```

```
-> address varchar(30),
```

```
-> age int
```

```
-> );
```

```
mysql>
```

n Ví dụ :

| Chèn các mẫu tin vào bảng Person

```
mysql> insert into Person values ('Thanh-Nghi', 'Do', '84/40, CMT8',31);
```

```
mysql> insert into Person values ('Nguyen-Khang', 'Pham', '43/20, Mau Than',27);
```

```
mysql> insert into Person values ('Nguyen-Binh', 'Le', '12, Nguyen Thong',18);
```

```
mysql> insert into Person values ('Trung-Tin', 'Nguyen', '31, Ngo Quyen',12);
```

```
mysql> insert into Person values ('Binh-Minh', 'Bui', 'C8, Truong Dinh',22);
```

```
mysql>
```

n Ví dụ :

| Thực hiện câu truy vấn trên bảng Person

```
mysql> select * from Person;
```

lastname	firstname	address	age
Thanh-Nghi	Do	84/40, CMT8	31
Nguyen-Khang	Pham	43/20, Mau Than	27
Nguyen-Binh	Le	12, Nguyen Thong	18
Trung-Tin	Nguyen	31, Ngo Quyen	12
Binh-Minh	Bui	C8, Truong Dinh	22

5 rows in set (0.00 sec)

n PHP nối kết đến MySQL

- | Tạo kết nối :

```
$conn = mysql_connect("ip_db_serv", "username", "passwd");
```

- | Chọn cơ sở dữ liệu để kết nối

```
$db = mysql_select_db("dbname", $conn);
```

- | Thực hiện câu SQL

```
$result = mysql_query("SQL command", $conn);
```

- | Lấy 1 dòng kết quả

```
$row = mysql_fetch_array($result);
```

- | Đọc giá trị một trường của mẫu tin

```
$val = $row["col-name"];
```

n PHP nối kết đến MySQL

- | Giải phóng tài nguyên của kết quả

```
mysql_free_result($result);
```

- | Đóng kết nối

```
mysql_close($conn);
```

```
<html>
```

```
<body>
```

```
<?php
```

```
$conn = mysql_connect("127.0.0.1", "nghi", "nghi")  
    or die("Could not connect: " . mysql_error());
```

```
$db = mysql_select_db("mydb", $conn)  
    or die("Could not select database");
```

```
$result = mysql_query("SELECT * FROM Person", $conn);
```

```
echo "<TABLE BORDER=1>";
```

```
echo "<TR><TH> LASTNAME </TH> <TH> FIRSTNAME </TH>  
    <TH> ADDRESS </TH> <TH> AGE </TH> </TR>";
```

```
while ($row = mysql_fetch_array($result)) {  
    echo "<TR>";  
    echo "<TD> " . $row["lastname"]. " </TD>";  
    echo "<TD> " . $row["firstname"]. " </TD>";  
    echo "<TD> " . $row["address"] . " </TD>";  
    echo "<TD> " . $row["age"] . " </TD>";  
    echo "</TR>";  
}  
  
echo "</TABLE>";  
?>  
  
</body>  
</html>
```

VÍ DỤ : PHP NỐI KẾT ĐẾN MYSQL



LASTNAME	FIRSTNAME	ADDRESS	AGE
Thanh-Nghi	Do	84/40, CMT8	31
Nguyen-Khang	Pham	43/20, Mau Than	27
Nguyen-Binh	Le	12, Nguyen Thong	18
Trung-Tin	Nguyen	31, Ngo Quyen	12
Binh-Minh	Bui	C8, Truong Dinh	22

❖ Cú pháp:

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

■ Hoặc:

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

❖ Cú pháp:

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

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

mysql_select_db("my_db", $con);

mysql_query("UPDATE Persons SET Age = '36'
WHERE FirstName = 'Do' AND LastName = 'Thanh-Nghi');

mysql_close($con);
?>
```

❖ Cú pháp:

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

XÓA DỮ LIỆU TRONG CSLD



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

mysql_select_db("my_db", $con);

mysql_query("DELETE FROM Person WHERE
LastName='Thanh-Nghi');

mysql_close($con);
?>
```

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

mysql_select_db("my_db", $con);

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

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

mysql_close($con);
?>
```

❖ Tạo HTML form để thêm mẫu tin mới tới bảng "Persons"

```
<html>
< body>
< form action="insert.php" method="post">
Firstname: <input type="text" name="txtFirstname" />
Lastname: <input type="text" name="txtLastname" />
Age: <input type="text" name="txtAge" />
<input name="txtAddress" type="text" id="txtAddress" />
< input type="submit" />
< /form>
< /body>
< /html>
```

- ❖ Khi người dùng click nút Submit trong HTML form, dữ liệu sẽ được gửi tới "**insert.php**".
- ❖ Tập tin "insert.php" nối kết CSDL và nhận giá trị từ form với các biến PHP **\$_POST**.
- ❖ Sau đó, hàm `mysql_query()` thực thi câu lệnh **INSERT INTO**, và mẫu tin mới sẽ được thêm tới bảng "Persons".


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

mysql_select_db("my_db", $con);

$sql="INSERT INTO Persons (FirstName, LastName, Age, Address)
VALUES
('$_POST[txtFirstname]',$_POST[txtLastname],$_POST[txtAge], '$_POST[txtAddress]')";

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

mysql_close($con)
?>
```

www.php.com

❖ Cung cấp:

- Thông tin về PHP như: phiên bản mới, các đặc trưng mới được hỗ trợ, các bản tin về PHP
- Download PHP
- Download các thư viện lập trình ứng dụng
- Download được các công cụ hỗ trợ, các trình tiện ích
- Tài liệu về PHP

❖ Sách:

- [David Sklar & Adam Trachtenberg, 2002] David Sklar & Adam Trachtenberg, “*PHP Cookbook*”, O'Reilly, 2002.
- [Simon Stobart & Mike Vassileiou, 2004] Simon Stobart & Mike Vassileiou, “*Php and Mysql Manual: Simple, Yet Powerful Web Programming*”, Springer Professional Computing, 2004.
- [Hugh E Williams & David Lane, 2002] Hugh E Williams & David Lane, “*Web Database Applications with PHP & MySQL*”, O'Reilly, 2002.

❖ Sách:

- [Luke Welling & Laura Thomson, 2001] Luke Welling & Laura Thomson, “*PHP and MySQL Web Development*”, SAMS, 2001.
- [Rasmus Lerdorf & Kevin Tatroe, 2002] Rasmus Lerdorf & Kevin Tatroe, “*Programming PHP*”, O'Reilly, 2002.
- [James Fuller et al., 2003] James Fuller, Ken Egervari, Harry Fuecks, Bryan Waters, Jon Stephens and Daniel Solin, “*Professional PHP Web Services*”, Wrox Press, 2003.

DHTML

JAVA
SCRIPT

LẬP TRÌNH
WEB

PHP

Cảm ơn!