

PHP for Base

GV: Bùi Quang Đăng

Contents

1

Introduction, Data types in PHP

2

Operators

3

Structured programming

4

PHP Form

PHP for Base

❖ Nội dung mã viết PHP

```
1  <!DOCTYPE html>
2  <html>
3  <body>
4
5  <?php
6
7      echo "Xin chao cac ban den voi PHP";
8      //Su dung comment tung dong
9
10     /*
11     Su dung de comment nhieu dong
12     */
13     ?>
14
15 </body>
16 </html>
```

PHP for Base

❖ Kiểu dữ liệu trong PHP

Type	Description
Boolean	Kiểu dữ liệu trả về TRUE hoặc FALSE
Integer	Kiểu số nguyên
Float	Kiểu số thập phân
String	Kiểu dữ liệu dạng chuỗi
Array	Kiểu dạng mảng
Object	Kiểu đối tượng
Resources	Là một dữ liệu đặc biệt, tham chiếu đến một nguồn bên ngoài. Được tạo và sử dụng bởi các function xác định. Resources xử các tập tin mở, kết nối database,...
NULL	Kiểu dữ liệu được trả về khi đối tượng không tồn tại, không xác định.

PHP for Base

❖ Kiểu dữ liệu trong PHP

- Để xác định dữ liệu thuộc kiểu gì trong PHP người ta có thể sử dụng hàm **var_dump()**;

Ví dụ:

```
$large_number = 2147483647;  
var_dump($large_number);           // int(2147483647)  
  
$large_number = 2147483648;  
var_dump($large_number);           // float(2147483648)  
  
$million = 1000000;  
$large_number = 50000 * $million;  
var_dump($large_number);           // float(500000000000)  
  
var_dump(25/7);                     // float(3.5714285714286)  
var_dump((int) (25/7));             // int(3)  
var_dump(round(25/7));              // float(4)
```

PHP for Base

❖ Operators

Operator	Description	Example
$x + y$	Phép cộng	$4 + 2 = 6$
$x - y$	Phép trừ	$4 - 2 = 2$
$x * y$	Phép nhân	$4 * 2 = 8$
x / y	Phép chia	$4/2 = 2$
$x \% y$	Phép chia lấy dư	$4\%2 = 0$
$x.y$	Ghép chuỗi	"Stan" . "ford" => "Stanford"

PHP for Base

❖ Operators

Operator	Description	Example
$x += y$	$x = x + y$	$x = 3, y = 5$
$x -= y$	$x = x - y$	
$x *= y$	$x = x * y$	
$x /= y$	$x = x / y$	
$x \% = y$	$x = x \% y$	
$x. = y$	$x = x . y$	
$++x$	Tăng trước khi trả về kết quả	
$x++$	Trả về kết quả sau đó mới tăng	
$--y$		
$y--$		

PHP for Base

❖ Operators

Operator	Description	Example
<code>x == y</code>	So sánh bằng	<code>9==10</code> return false
<code>x != y</code>	So sánh không bằng	<code>9!=10</code> return true
<code>x <> y</code>	x khác với y	<code>9 <> 10</code> return true
<code>x > y</code>		
<code>x >= y</code>		
<code>x < y</code>		
<code>x <= y</code>		
<code>x and y</code>	True nếu cả 2 đều đúng	
<code>x or y</code>	True nếu một hoặc cả hai đều đúng	<code>x=3; y=6</code> <code>(x==3 or y==6)</code> return true

PHP for Base

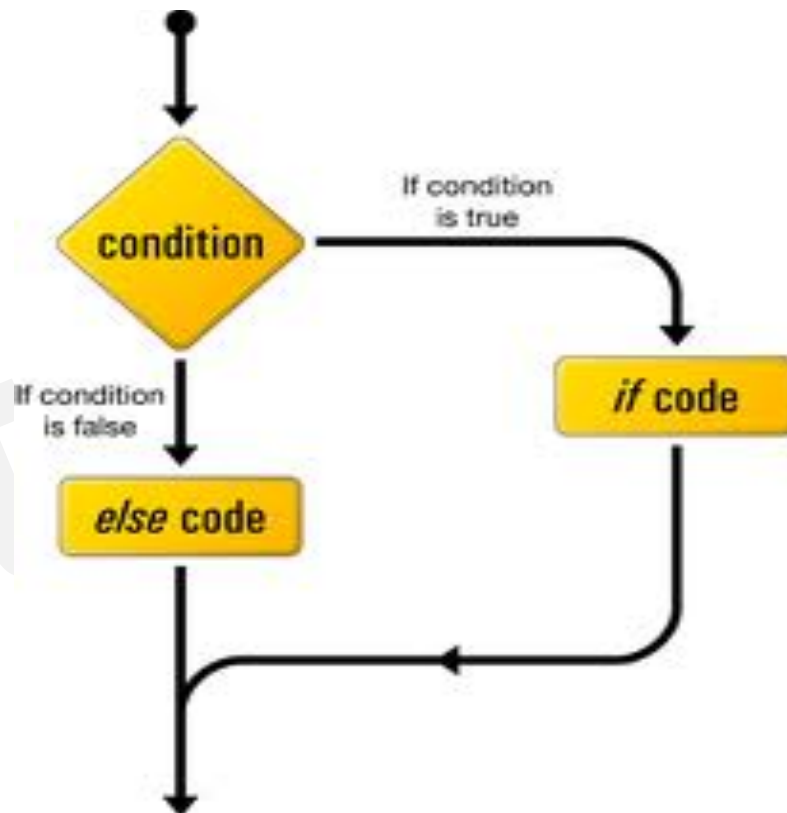
❖ Operators

Operator	Description	Example
<code>x && y</code>	True nếu cả hai đều đúng	<code>x=3; y=5</code> <code>(x==3 && y==5) return</code> True
<code>x y</code>	True nếu mỗi hoặc cả 2 đều đúng	
<code>x xor y</code>	True nếu x hoặc y đúng	
<code>! x</code>	True nếu x không đúng	

PHP for Base

❖ Statement

▪ If...Else



PHP for Base

❖ Statement

▪ If...Else

If(biểu thức điều kiện)

{

//Thực hiện công việc nếu điều kiện đúng

}

else

{

//Thực hiện công việc này nếu điều kiện sai

}

PHP for Base

❖ Statement

▪ If...Else

```
$a = 6; $b=2; $tong=0;  
If($b>0)  
{  
    $chia = $a / $b;  
}  
else  
{  
    echo "Bạn cần nhập số b lớn hơn 0";  
}
```

PHP for Base

❖ Statement

▪ If...Else

If(biểu thức điều kiện)

{

 if(biểu thức điều kiện) {

 //Thực hiện công việc

 }

}

else

{

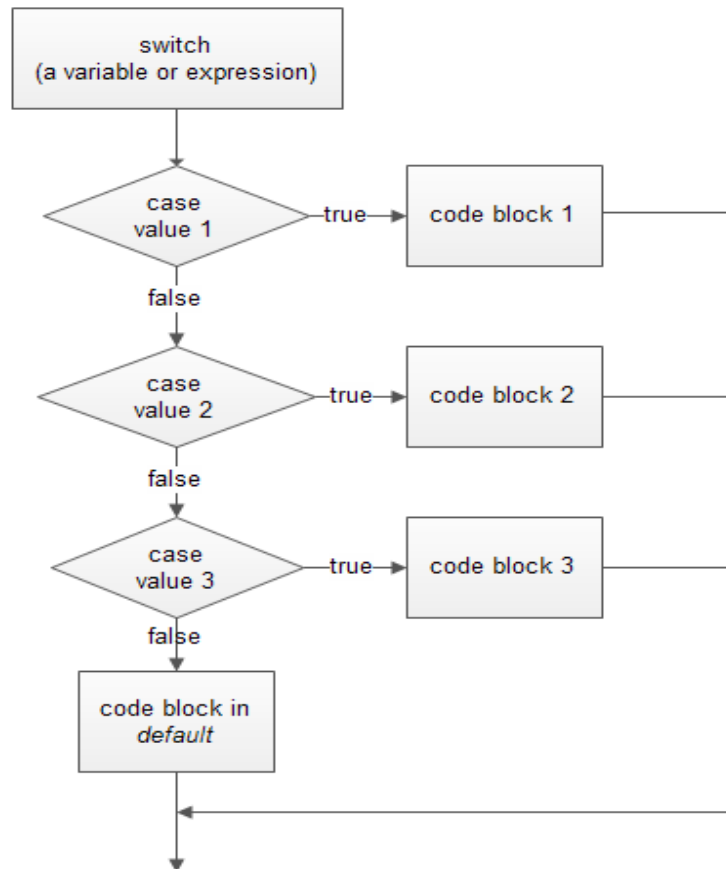
 //Thực hiện công việc này nếu điều kiện sai

}

PHP for Base

❖ Statement

▪ Switch



PHP for Base

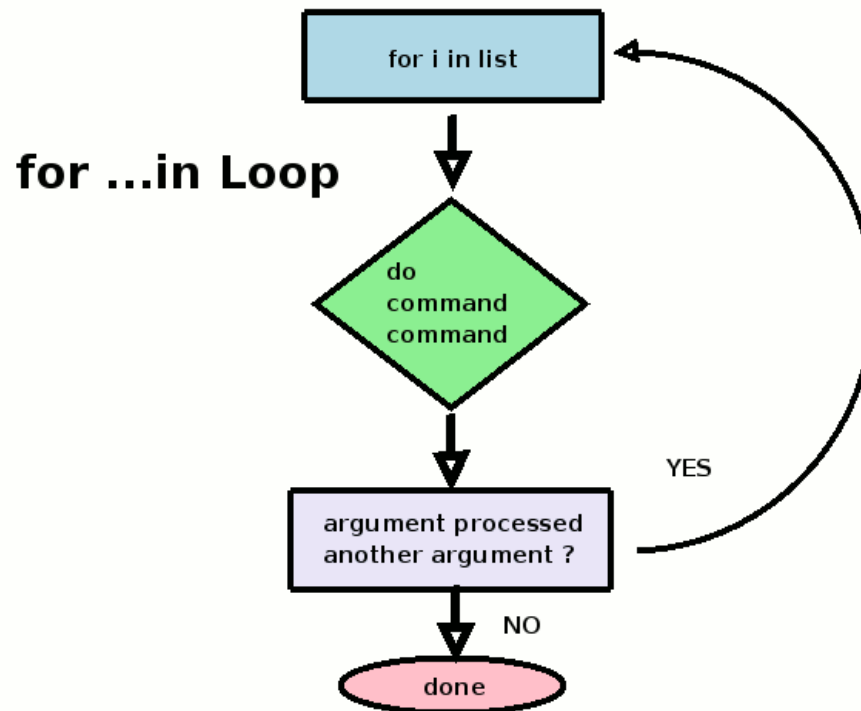
❖ Statement

▪ Switch

```
switch(n)
{
    case giatri1: //Thực hiện công việc 1
        break;
    case giatri2: //Thực hiện công việc 2
        break;
    ...
    default: //Thực hiện công việc mặc định
        break;
}
```

PHP for Base

❖ PHP Loops



PHP for Base

❖ PHP Loops

▪ While

- Thực hiện công việc trong vòng lặp nếu điều kiện đúng

- **Cú pháp:**

While (điều kiện)

```
{  
    //Thực hiện công việc  
}
```

PHP for Base

❖ PHP Loops

▪ Do...While

- Thực hiện công việc trong vòng lặp sau đó mới kiểm tra điều kiện của vòng lặp => **Luôn thực hiện ít nhất một lần**
- **Cú pháp:**

Do

{

//Thực hiện công việc

}

While (điều kiện)

PHP for Base

❖ PHP Loops

▪ for

- Thực hiện công việc trong vòng lặp theo biểu thức điều kiện với số bước lặp được xác định.
- **Cú pháp:**

```
for(khởi tạo; biểu thức điều kiện; bước lặp)
{
    //Thực hiện công việc
}
```

PHP for Base

❖ PHP Loops

▪ for

- Ví dụ:

```
for($i=0; $i<5; $i++)  
{  
    echo "Chào mừng bạn đến với PHP " . $i;  
}
```

PHP for Base

❖ PHP Loops

▪ foreach

- Thực hiện công việc trong vòng lặp, duyệt từ đầu đến cuối mảng, danh sách.

- **Cú pháp:**

```
Foreach($arr as $value)
```

```
{
```

```
    //Thực hiện công việc
```

```
}
```

PHP for Base

❖ PHP Loops

▪ foreach

- Ví dụ:

```
<?php
```

```
$arr = array("Stanford", "Dạy kinh nghiệm", "lập trình");
```

```
Foreach($arr as $value)
```

```
{
```

```
    echo $value . "<br/>";
```

```
}
```

```
?>
```

PHP for Base

PHP FORM

PHP for Base

❖ PHP Form

- Trong một trang PHP để lấy được thông tin trên các đối tượng như textbox, checkbox, radio,...thì người dùng cần khai báo trong cặp thẻ **<form></form>**
- Các thuộc tính của thẻ form hay được sử dụng như sau:
 - **action**: Khai báo địa chỉ tên trang gửi thông tin khi thực hiện submit.
 - **method**:
 - **GET**: Yêu cầu dữ liệu từ một nguồn xác định
 - **POST**: Submit dữ liệu từ một nguồn xác định
 - **enctype**: multipart/form-data sử dụng để xử lý khi gửi dữ liệu, tải file

PHP for Base

❖ PHP Form

- Sử dụng để lấy thông tin từ người dùng nhập vào qua các đối tượng input.
- Dùng 2 đối tượng **\$_GET** và **\$_POST** để lấy thông tin trong PHP:
 - **\$_GET**: Thông tin được truyền qua địa chỉ URL trong PHP với method của Form là GET.
 - **\$_POST**: Để lấy thông tin từ form người dùng gửi lên khi submit với method của Form là POST.

PHP for Base

❖ PHP Form

- **\$_GET**: Là một biến trong PHP sử dụng để lấy thông tin các các thẻ trong form được gửi trên URL qua phương thức GET.

Ví dụ: `<body>`

```
<form action="checklogin.php" method="GET">
Username: <input type="text" name="txtUserName"/><br/>
Password: <input type="password" name="txtPassword"/><br/>
<input type="submit" value="Login"/>
</form>
</body>
```

checklogin.php

```
Username: <?php echo $_GET["txtUserName"]; ?><br/>
Password: <?php echo $_GET["txtPassword"]; ?><br/>
```

PHP for Base

❖ PHP Form

- **\$_POST**: Là một biến trong PHP sử dụng để lấy thông tin từ form được gửi qua phương thức POST.

Ví dụ:

Họ tên:

Điện thoại:

Email:

UserName:

Password:

Địa chỉ:

```
<?php
try{
include ("connect.php");

$userid = (int)$_GET['userid'];

$hoten = $_POST['hoten'];

$username= $_POST['username'];

$dienthoai= $_POST['dienthoai'];
```

PHP for Base

❖ PHP Form

▪ **\$_REQUEST:**

- Là một biến HTTP Request nó chứa các nội dung của \$_GET, \$_POST and \$_COOKIE.
- Sử dụng được cả 2 method GET và POST

Ví dụ:

```
<?php
if(isset($_REQUEST['userid']) && isset($_REQUEST['kq']))
{
    echo '<center><font color="#FF0000">Cap nhat thanh Cong</font></center>';
}
?>
```



Thank You !