**PHP - Hypertext Preprocessor**

**1. Tổng quan**

* Khai báo vùng viết mã PHP

<?php  
 // PHP code goes here  
 ?>

* Chú thích trong php: // hoặc /\*..\*/ hoặc #
* Hiển thị giá trị ra trình duyệt

Echo” thông tin cần hiển thị”;

Print” thông tin cần hiển thị”; //chỉ in 1 chuỗi

Var\_dump(“thong tin”); //in kèm data type

+ Kết thúc một dòng lệnh: “;”

+ Thông tin bên trong dấu **“”** có thể là một chuỗi, một biến hoặc các thẻ HTML

+ *Nếu muốn sử dụng nhiều cặp dấu* ***“”*** *lồng nhau thì các cặp dấu* ***“”*** *bên trong phải được khóa bởi các lý tự* ***\****.* ***VD: echo “<p align=\”center\”>Hello Word</p>”;***

* Biến, hằng, các kiểu dữ liệu

+ Khai báo biến: $ten\_bien;

+ Khai báo hằng:

**define (string tên\_hằng, giá\_trị\_hằng );**

+ Chuỗi trong PHP

+ Nối chuỗi: “.”

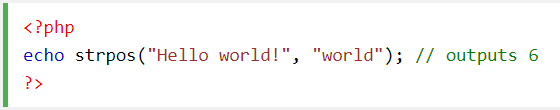
+ Các kiểu dữ liệu trng PHP: có 6 kiểu chính Integer, double, string, boolean, object, array.

+ Kiểm tra kiểu dữ liệu của biến: **gettype(ten\_bien\_can\_kiem\_tra);**

1. **String**

* **Strlen() : trả về độ dài chuỗi**
* **Str\_word\_count() : đếm số lượng tử trong một chuỗi**
* **Strrev(): đảo ngược một chuỗiChú thích trong PHP: sử dụng ‘//’ hoặc /\*..\*/ hoặc “#”**
* **Strpos(): tìm kiếm một văn bản trong chuỗi, trả về vị trí của kí tự đầu tiên hoặc false.**

**Ex:**



* **Str\_replace(): thay thế một kí tự bằng một kí tự khác trong chuỗi.**

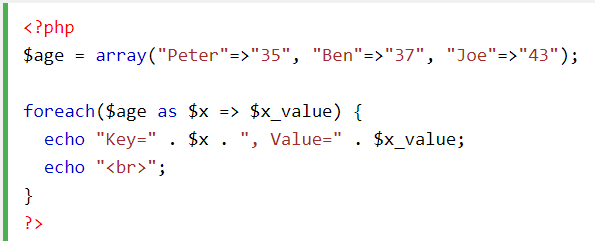


1. **Number**

* **Interger: function check type-> is\_int(), is\_interger(), is\_long**
* **Float: function check type-> is\_float(), is\_double()**
* **Infinity (vô cực): check type-> is\_finite(), is\_infinite()**
* **NaN: check if a value is not a number: is\_nan()**
* **Numerical strings: is\_numeric() function tìm xem một biến có phải số hay không, return true/false**
* **Ép kiểu: (int)$x**

1. **Array**

* **Khai báo: array();**
* **Associative array:**



* Mảng đa chiều:

$cars = array (

array("Volvo",22,18),

array("BMW",15,13),

array("Saab",5,2),

array("Land Rover",17,15)

);

* Sắp xếp mảng:

Sort(): sx mảng theo chiều tăng dần

Rsort(): sx mảng theo chiều giảm dân

Asort(): sx mảng kết hợp theo chiều tăng dần, theo value

Ksort(): sx mảng kết hợp theo chiều tăng dần, theo key

Arsort(): sx mảng kết hợp theo chiều giảm dần, theo value

Krsort(): sx mảng kết hợp theo chiều giảm dầ, theo key

* Toán tử trong PHP

+ Toán tử gán: “=“

+ Toán tử số học: +,-,\*,/,%...

+ Toán tử so sánh: ==, !=, === (đồng nhất), >, >=, <, <=.

+ Toán tử logic: ||, &&, !, or, xor, and,...

+ Toán tử kết hợp: ++, --, +=, -=, \*=, /=.

* Biểu thức điều kiện, vòng lặp

+ if, if...else, switch...case,..

+ for, while, do...while,...

PHP\_02

1. Các phương thức truyền dữ liệu trong PHP
   1. Truyền và nhận biến qua FORM



* Các thuôc tính quan trọng của form:

+ name: tên form

+ method: phương thức truyền dữ liệu qua form, 2 phương thức chính POST (chủ yếu) &GET

+ action: địa chỉ của trang xử lý dữ liệu được gửi từ form

* Cách thức truyền và nhận một biến qua FORM bởi các phương thức truyền biến POST&GET

**Cú pháp:**

**POST ta có: $\_POST[“name”].** Với name chính là tên của trường nhập dữ trong Form

**GET ta có: $\_ GET[“name”].** Với name chính là tên của trường nhập dữ trong Form

* 1. Truyền và nhận biến qua trình duyệt
* Truyền và nhận biến qua trình duyệt bởi phương thức GET

*Truyền biến bằng phương thức GET qua trình duyệt là truyền một chuỗi ký tự tiếp sau phần mở rộng của file .php. Chuỗi này bao gồm tên biến (một hoặc nhiều biến) và giá trị tương ứng của các biến đó*

**Cú pháp:**

**Filename.php?ten\_bien=gia\_tri:** Với **?page=giatri** là chuỗi ký tự được thêm vào sau phần mởi rộng .php để truyền một biến có tên là ten\_bien có giá trị là gia\_tri

**$\_ GET[“name”].** Với name chính là tên của biến được truyền trên trình duyệt

* *Chúng ta có thể dễ dàng truyền đồng thời nhiều biến qua trình duyệt và nhận chúng bằng phương thức GET*

**Cú pháp:**

**Filename.php?ten\_bien\_1=gia\_tri&ten\_bien\_2=gia\_tri…ten\_bien\_n=gia\_tri:**

1. Dữ liệu kiểu mảng và các hàm hỗ trợ mảng trong PHP
   1. Làm việc với mảng

* Định nghĩa: *Mảng trong php* *là một danh sách các phần tử có hoặc không cùng kiểu dữ liệu. Mảng có thể là mảng một chiều hay nhiều chiều. Mảng  có 2 thành phần là chỉ mục (key) và giá trị (value). Chỉ mục (key) có thể là số nguyên hoặc là chuỗi*

**Cú pháp khai báo một mảng (Cách 1)**

**$mang = array(phan\_tu\_1, phan\_tu\_2,…, phan\_tu\_n);**

*Với phan\_tu\_1 (phần tử đầu tiên trong mảng) có chỉ mục (key) là 0 và giá trị (value) là phan\_tu\_1*

**Cú pháp khai báo một mảng (Cách 2)**

**$mang = array();**

**$mang[] = phan\_tu\_1;**

**$mang[] = phan\_tu\_1;**

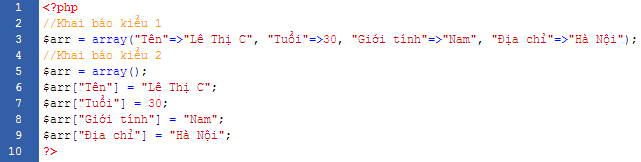
**…**

**$mang[] = phan\_tu\_n;**

*Với phan\_tu\_1 (phần tử đầu tiên trong mảng) có chỉ mục (key) là 0 và giá trị (value) là phan\_tu\_1*

* Mảng kết hợp trong PHP

*Mảng kết hợp là mảng chứa các phần tử có các chỉ mục (key) là các chuỗi*



* Thêm một phần tử mới vào mảng: **$bien\_mang[chi\_muc] = gia\_tri;**
* Xuất giá trị của một phần tử trong mảng: **$bien\_mang[chi\_muc];**