

HỌC PHẦN “LẬP TRÌNH WEB VỚI PHP”

(Web programming with PHP)

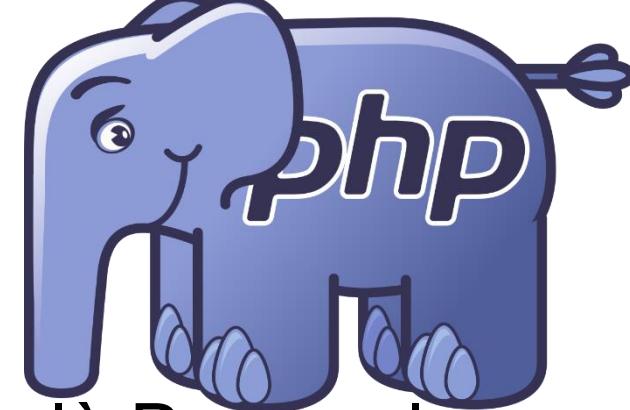
GVGD: ThS. Trần Mạnh Đông

PHP cơ bản

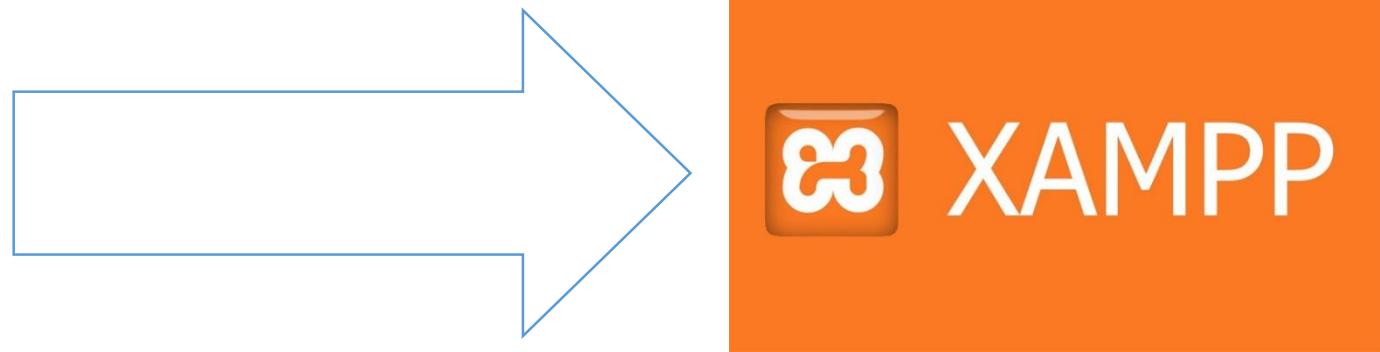
NỘI DUNG (1)

- Nhúng PHP vào HTML
- Chú thích, câu lệnh
- Kiểu dữ liệu, biến và hằng

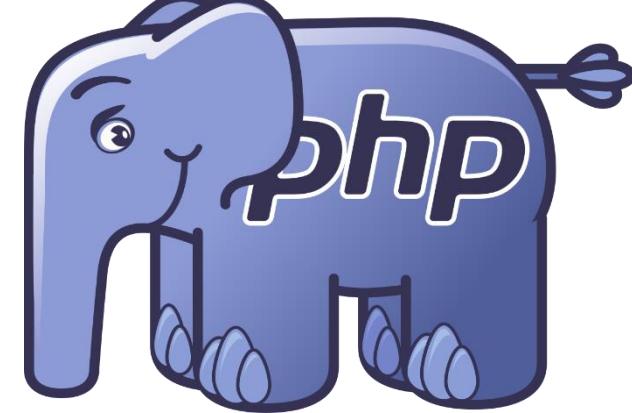
Nhắc lại: Giới thiệu về PHP



- PHP (Viết tắt PHP Hypertext Preprocessor, tên gốc là Personal Home Page): Là ngôn ngữ lập trình viết web động (dynamic web)
 - Bộ biên dịch PHP là mã nguồn mở
 - Là ngôn ngữ server-side script thực thi phía web server (tương tự như: ASP.NET, JSP, ...)
 - Thường kết nối với hệ quản trị CSDL MySQL
- Để thiết kế web động với PHP và MySQL, cần cài đặt:
 - Web server: Apache
 - PHP
 - Hệ quản trị CSDL MySQL



Nhắc lại: Giới thiệu về PHP



- Các điểm nổi bật của PHP
 - PHP dễ học, dễ sử dụng
 - Phát triển web hiệu quả
 - Tương thích với nhiều hệ điều hành phổ biến hiện nay
 - Tương tác hầu hết Hệ quản trị CSDL có sẵn
 - Thể hiện được tính bền vững, chặt chẽ, linh hoạt
 - Tiết kiệm chi phí

Nhúng PHP vào HTML

- Các tệp tin (file) PHP được lưu với phần mở rộng “**.php**”
- Để thêm mã PHP: sử dụng thẻ bắt đầu với **<?php** và kết thúc là **?>**.
- Khi viết mã PHP, thường mở thẻ PHP trước thẻ HTML đầu tiên.
- Thẻ PHP chứa các lệnh xử lý một số thao tác ban đầu và khởi tạo những biến sử dụng sau này.
- Sau đó, các thẻ PHP ngắn được sử dụng để hiển thị tại các phần khác nhau của tài liệu HTML

Cú pháp (syntax)

- Cú pháp (*syntax*) của PHP là những quy tắc người lập trình phải tuân thủ khi viết mã
 - Cú pháp PHP khá giống C/C++, Java, JavaScript
 - Câu lệnh PHP kết thúc bằng dấu “;”
 - PHP bỏ qua mọi khoảng trắng thừa (dấu cách – space, ký tự nhiều dấu cách – tab, ký tự xuống dòng) trong câu lệnh

Chú thích (comment)

- Chú thích (comment) sẽ giúp giải thích nội dung của dòng lệnh/khoi lệnh
- Chú thích sẽ được bỏ qua khi PHP engine xử lý mã
- Chú thích được viết trong code với ký tự: **//**, **#** hoặc cặp **/* .. */**

```
<?php
    //Đây là ví dụ về ghi chú trong PHP - ghi từng dòng
    /* - Các ghi chú không được thực thi.
       - Chúng nhằm mục đích làm rõ/giải thích thêm về mục đích của
       biến, hàm, lớp,...*/
    # APACHE bỏ qua các ghi chú trong PHP
    echo "<h1>";
    echo "<h1>Ví dụ về Ghi chú (Comment) trong PHP!</h1>";
    echo "</h1>";
?>
```

Kiểu dữ liệu trong PHP

- Các kiểu dữ liệu cơ bản trong PHP:

- Kiểu số nguyên (integer data type): được dùng để lưu số tự nhiên
- Kiểu số thực double (double data type): lưu các số thực với phần thập phân sau dấu chấm “.”
- Kiểu boolean (boolean data type): lưu các giá trị thuộc một trong hai trạng thái true/false, yes/no, on/off. Để biểu thị dữ liệu boolean, viết mã là TRUE hoặc FALSE (không phân biệt chữ hoa/thường).
- Kiểu chuỗi (string data type): Lưu các đoạn văn bản có thể chứa bất kỳ ký tự nào. Để viết giá trị chuỗi, đặt chuỗi vào trong cặp dấu nháy đơn ‘ ’ hoặc cặp dấu nháy kép “ ”
- Kiểu mảng (array): bộ chứa nhiều giá trị thuộc một hoặc nhiều kiểu dữ liệu
- Kiểu tham chiếu đối tượng (object): bộ chứa dữ liệu (thuộc tính) và hàm (phương thức)

Kiểu dữ liệu trong PHP...

- Giá trị nguyên

25 // số nguyên
-25 //sô nguyên âm

- Giá trị double

25.15 //giá trị có dấu chấm động
-125.15 //giá trị có dấu chấm động âm
5.7e9 //tương đương với 5700000000
3.5e-9 //tương đương với 0.000000035
-4.7e9 //tương đương với -4700000000

- Giá trị Boolean

true //tương ứng với đúng, có (yes) hoặc bật (ON)
false //tương ứng với sai, không (no) hoặc tắt (OFF)

- Giá trị chuỗi

‘Lập trình PHP’ //chuỗi với nháy đơn
“Lập trình PHP” //chuỗi với nháy kép
‘ ’ //chuỗi rỗng
null //giá trị NULL

Biến (variables) và hằng (constant)

- Biến được dùng để lưu trữ những dữ liệu có thể thay đổi khi ứng dụng chạy
- Khai báo biến
 - Cú pháp: **\$ten_bien**
- Các biến trong PHP đều bắt đầu bằng ký hiệu đô-la “\$”
- Để gán giá trị cho biến, sử dụng toán tử gán “=”, sau là biểu thức trả về
 - Cú pháp: **\$ten_bien=<gia_tri>;**

Biến (variables) và hằng (constant)...

- Ví dụ

```
$count = 10;                      // số nguyên nguyên bản
$list_price = 9.50;                // số double nguyên bản
$first_name = 'Bob';               // chuỗi nguyên bản - sử dụng dấu nháy đơn
$first_name = "Bob";               // chuỗi nguyên bản - sử dụng dấu nháy kép
$is_valid = false;                 // Boolean nguyên bản - viết thường
$product_count = $count;          // $product_count = 10;
$price = $list_price;              // $price = 9.50
$name = $first_name;              // $name = "Bob"
$is_new = $is_valid;              // $is_new = FALSE
```

Biến (variables) và hằng (constant)...

- Quy tắc đặt tên biến trong PHP:
 - Tên biến có phân biệt chữ hoa, chữ thường
 - Tên biến có chứa chữ cái, số và ký tự gạch dưới “_”
 - Tên biến không chứa ký tự đặc biệt
 - Tên biến không bắt đầu bằng số hay hai ký tự gạch dưới “_”
 - Tên biến không được trùng với các từ khóa trong PHP (VD: \$this)

Biến (variables) và hằng (constant)...

- **Biến cục bộ (local variable):**

- Biến được khai báo trong hàm
- Khi ra khỏi hàm, biến cục bộ và giá trị của nó sẽ bị hủy bỏ
- Lưu ý: Khi có cùng tên biến thì tên biến cục bộ và biến ngoài hàm/biến toàn cục là hai biến hoàn toàn khác nhau

```
<?php
function myTest()
{
    $x=2004;// phạm vi cục bộ
    echo $x;
}
myTest(); //kết quả in ra 2004
echo '<br>';
echo $x; //Không có
?>
```

Biến (variables) và hằng (constant)...

- **Biến toàn cục (global variable):**

- Có thể truy xuất bất cứ nơi nào trong trang
- Khi muốn sử dụng và cập nhật biến toàn cục trong hàm thì phải:
 - Dùng từ khóa “**global**” phía trước biến hoặc
 - Dùng **\$_GLOBALS['ten_bien']**

```
<?php
$x = 5;
$y = 10;
function sum()
{
    global $x, $y;
    $y = $x + $y;
}
sum();
echo $y;//kết quả in ra 15
?>
```

Biến (variables) và hằng (constant)...

- **Biến tĩnh “static”:**

- Không mất đi giá trị ra khỏi hàm
- Sẽ giữ nguyên giá trị trước đó khi hàm được gọi một lần nữa
- Khai báo từ khóa “**static**” trước tên biến

```
<?php
function myTest()
{
    static $x=2004;// biến static
    echo $x;
    $x++;
}
myTest();// kết quả in ra 2004
myTest(); // kết quả in ra 2005
myTest();// kết quả in ra 2006
?>
```

Biến (variables) và hằng (constant)...

- Hằng (constant): là biến có giá trị luôn luôn không đổi
- Quy ước đặt tên hằng: Thường viết hoa cho toàn bộ tên hằng
- Khai báo hằng:

```
Definal('MAX_VALUE', 10)//Hằng số nguyên  
Define('PI',3.14)//Hằng số thực  
Definal('EAUT', 'TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á')//Hằng chuỗi
```

Biến (variables) và hằng (constant)...

- Hàm **define()**: là hàm của PHP dùng để khai báo, định nghĩa cho một hằng số

```
<?php
    define("EAUT", "TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á!");
    define("PI", 3.14);

    echo EAUT. "<BR />";
    echo PI;
?>
```

- Hàm **defined()** là hàm dùng để Kiểm tra xem một hằng số được đặt tên đã cho có tồn tại hay không

Biến (variables) và hằng số (constant)...

- **Sự khác nhau giữa hằng số (constant) và biến (variable):**
 - Tên hằng số không cần bắt đầu bằng ký tự \$ như biến
 - Hằng số không thể sử dụng phép gán giá trị như biến (=), giá trị của nó phải được xác định duy nhất tại hàm **define**.
 - Hằng số có thể truy cập bất cứ ở đâu mà không giới hạn phạm vi như biến
 - Ngay sau khi giá trị hằng thiết lập, nó không bị thay đổi hay định nghĩa lại nữa

Demo

- Nhúng PHP vào mã HTML
- Các ghi chú trong PHP
- Biến và Kiểu dữ liệu trong PHP