

LAB 2: PHP CĂN BẢN

1. Mục tiêu

- Biết cách chạy và dịch một file php.
- Hiểu được cách thức biên dịch một file PHP
- Biết cách sử dụng biến, hằng số.
- Phân biệt các kiểu dữ liệu trong PHP
- Vận dụng được các toán tử, hàm liên quan đến số, chuỗi.
- Phân biệt và vận dụng các hàm include, require, include_once, require_once.

2. Tóm tắt lý thuyết

- Biến trong PHP được bắt đầu bằng ký tự \$ và theo sau là tên biến. Ví dụ: \$a.
- Nguyên tắc đặt tên biến: tên biến bắt đầu bằng ký tự chữ cái hoặc ký tự _ , không được bắt đầu bằng ký số.
- Ví dụ:
 - o \$chuoi, \$_123, \$x1: là những biến hợp lệ.
 - o \$1x, \$34: là những biến không hợp lệ.
- **Tên biến có phân biệt chữ hoa chữ thường.**
- Biến tĩnh là biến được khai báo bên trong 1 hàm và sau từ khóa static.
- Hằng số được khai báo như sau: define(“tên_hằng”, giá_trị)
- Các kiểu dữ liệu trong PHP:
 - o Kiểu luận lý(Boolean)
 - o Số nguyên (Integer)
 - o Số thực (float/double)
 - o Kiểu chuỗi (string)
 - o Kiểu mảng (array).
- Các phép toán số học: + (cộng), - (trừ), * (nhân), / (chia), % (chia dư).
- Các phép toán so sánh: ==, ===, >, >=, <, <=, != hoặc <>
- Các hàm hay sử dụng: include, require, include_once, require_once, isset, unset
- Cấu trúc điều khiển:

Cú pháp 1

if (biểu thức điều kiện)

{

 Các lệnh được thực hiện nếu điều kiện thỏa

}

Cú pháp 2

if (biểu thức điều kiện)

{

```

        các lệnh được thực hiện nếu điều kiện thỏa
    }
else
{
    các lệnh được thực hiện nếu điều kiện không thỏa
}

```

Cú pháp 3

```

if (điều kiện 1)
{
    các lệnh được thực hiện nếu điều kiện 1 thỏa
}
elseif (điều kiện 2)
{
    các lệnh được thực hiện nếu điều kiện 2 thỏa
}
else
{
    các lệnh được thực hiện nếu tất cả các điều kiện trên không thỏa
}

```

Switch ...case... // biểu thức của chọn 1 trong nhiều điều kiện

switch (biến điều kiện)

```

{
    case <giá trị1>:
        các lệnh được thực hiện nếu thỏa giá trị 1
        break;
    case <giá trị2>:
        các lệnh được thực hiện nếu thỏa giá trị 2
        break;
    .....
    default:
        Khối lệnh được thực hiện khi tất cả các trường hợp đều
        không thỏa
}

```

}

3. Ví dụ

- 3.1 Double click vào file lab2_1a.html. Cho biết kết quả.
- 3.2 Double click vào file lab2_1b.php. Cho biết kết quả.
- 3.3 Chép thư mục lab 2 vào thư mục gốc web server và chạy file lab2_1b.php từ localhost. Cho biết kết quả.
- 3.4 Mở file lab2_1a.html và lab2_1b.php để xem code. Sau đó xem source code file lab2_1b.php trên trình duyệt.
- 3.5
- 3.6 Ví dụ lab2_3.php: cách sử dụng hằng số
- 3.7 Ví dụ lab2_4.php: toán tử so sánh.
- 3.8 lab2_5 ví dụ về sử dụng hàm isset để kiểm tra biến có tồn tại không và cách gọi 1 file php khác.

4. Vận dụng

- 4.1 Trong lab2_2.php, xóa bỏ global thì kết quả xuất ra mà hình là gì? Vì sao?
- 4.2 Có thể thay thế dòng 10: `const PI = 3.14;` bằng lệnh nào khác không?
- 4.3 Trong lab2_4.php, biến a kiểu số nguyên, biến x kiểu chuỗi nhưng `a+x` hay `x+a` đều được do PHP tự ép kiểu. Vậy tại sao phép so sánh dòng 21 lại xác định a và x khác nhau? Muốn xuất ra “a và x giống nhau” thì phải sửa gì dòng 21.
- 4.4 Comment dòng 10 và chạy lab2_5.php. Giải thích kết quả.
- 4.5 Thêm `include("lab2_5b.php");` vào lab2_5.php (không comment dòng 10) tại dòng 11 và lưu lại thành tên **vd4_5.php**. Chạy **vd4_5.php** và cho biết kết quả.
- 4.6 Thêm `include("lab2_5b.php");` vào **vd4_5.php** tại dòng 12 và lưu thành tên **vd4_6.php**. Chạy **vd4_6.php** và cho biết kết quả.
- 4.7 Thay `include("lab2_5b.php");` dòng 12 trong file **vd4_6.php** thành `include_once("lab2_5b.php");` và lưu lại với tên **vd4_7.php**. Chạy **vd4_7.php** và so sánh kết quả với kết quả câu 4.6.
- 4.8 Lần lượt thay `include` thành `require` trong **vd4_6.php** và **vd4_7.php** và lưu lại với tên **vd4_6b.php** và **vd4_7b.php**. So sánh kết quả với câu 4.6 và 4.7.
- 4.9 Xóa file **lab2_5b.php** rồi lần lượt chạy **vd4_6.php** và **vd4_6b.php**. So sánh kết quả, từ đó rút ra sự khác nhau giữa `include` và `require`.

5. Nâng cao

- 5.1 Khai báo 2 biến số nguyên \$a\$ và \$b\$. Cho biết phần nguyên và phần dư của \$a/b\$ là bao nhiêu?
- 5.2 Giả sử có biến \$a\$ chứa giá trị kiểu số (số nguyên hoặc số thực). Viết code để kiểm tra xem \$a\$ là số nguyên hay số thực.
- 5.3 Giải phương trình bậc 2: $ax^2 + bx + c = 0$.