

**ỦY BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG CAO ĐẲNG KỸ THUẬT LÝ TỰ TRỌNG TP. HCM**

-----oOo-----

**ĐỀ TÀI SÁNG KIẾN KINH NGHIỆM
NĂM HỌC 2015 - 2016**

XÂY DỰNG HỆ THỐNG BÀI TẬP LẬP TRÌNH PHP

BIÊN SOẠN ĐỀ TÀI: Trần Việt Khánh

Lời Giới Thiệu

Lập trình PHP là học phần thuộc khối kiến thức chuyên ngành đối với các sinh viên thuộc các chuyên ngành công nghệ thông tin. Lập trình PHP cũng là khối kiến thức quan trọng mà sinh viên cần phải có được để trang bị hành trang làm việc sau khi tốt nghiệp đáp ứng nhu cầu trong thị trường công nghệ thông tin ngày nay.

Hệ thống bài tập lập trình PHP này trình bày các chủ đề bài tập liên quan đến ngôn ngữ lập trình PHP và cơ sở dữ liệu MySQL. Mỗi chủ đề được thiết kế bao gồm các phần: Tóm tắt lý thuyết, hướng dẫn giải một số dạng bài tập cơ bản và một số bài tập chọn lọc nâng cao. Trong mỗi chủ đề đều có các ví dụ minh họa và mỗi ví dụ đều có trình bày về code mẫu, kết quả đạt được khi chạy ứng dụng Web. Phần cuối của hệ thống bài tập lập trình PHP có hướng dẫn giải cho một số bài tập tiêu biểu, đồng thời bổ sung một số bài tập tổng hợp, nâng cao để sinh viên tham khảo và tự rèn luyện kỹ năng phân tích vấn đề, rèn luyện tư duy và phong cách lập trình.

Hệ thống bài tập lập trình PHP này được biên soạn để giảng dạy cho học phần lập trình PHP ở bậc cao đẳng. Hệ thống bài tập này cũng có thể làm tài liệu tham khảo cho các giáo viên đang giảng dạy kiến thức lập trình PHP ở các bậc đào tạo khác, và giúp cho sinh viên tự học có kết quả. Tùy theo nội dung chương trình đào tạo mà giáo viên có thể chọn một số nội dung phù hợp trong hệ thống bài tập này để giảng dạy.

Tôi trân trọng giới thiệu với bạn đọc hệ thống bài tập lập trình PHP này và hy vọng rằng nó sẽ giúp cho việc giảng dạy và học tập môn lập trình PHP được thuận lợi hơn.

Cuối cùng, chúng tôi xin gửi lời cảm ơn chân thành đến các đồng nghiệp trong khoa công nghệ thông tin trường Cao Đẳng Kỹ Thuật Lý Tự Trọng TP. HCM đã có nhiều đóng góp quý báu, đã cùng tôi chia sẻ nội dung tập bài giảng lập trình PHP này trong suốt thời gian qua.

Thành phố Hồ Chí Minh, ngày 11 tháng 11 năm 2015

Trần Việt Khánh

Chương 1

CÁC KHÁI NIỆM PHP CƠ BẢN

A. TÓM TẮT LÝ THUYẾT

1.1. Giới thiệu

- ✓ PHP với tên gốc là Personal Home Page. Là ngôn ngữ mã nguồn mở để viết các trang web động.
- ✓ Apache: máy chủ web để chạy các ứng dụng PHP sau khi lập trình.
- ✓ Kết hợp với hệ quản trị CSDL MySQL

1.2. Các thành phần cơ bản PHP

- ✓ Nhúng PHP vào HTML, có thể nhúng mã PHP vào mọi vị trí trong trang HTML.
- ✓ Các đoạn mã PHP phải được nằm trong thẻ mở `<?php` và thẻ đóng `?>`

`<?php`

`// Đoạn lệnh PHP ở đây`

`?>`.

- ✓ Một cấu trúc lệnh thông thường của PHP có thể được tách làm nhiều phần, mỗi phần đặt giữa `<?php ... ?>`
- ✓ Đặc điểm của PHP là có khả năng hướng đối tượng, phân biệt chữ hoa/chữ thường, mỗi lệnh kết thúc bởi chấm phẩy (;)
- ✓ Chú thích trong PHP, chú thích một dòng lệnh (// Dòng chú thích), chú thích trên một đoạn lệnh (/* Đoạn chú thích */)

1.3. Biến, hằng, các toán tử

✓ Biến

- Bắt đầu bằng dấu đô la (\$), tiếp ngay sau \$ là tên biến.
- Tên biến bao gồm chữ cái, chữ số, dấu gạch nối (_) và phải bắt đầu bằng chữ cái hoặc dấu gạch nối.
- Ví dụ: \$a, \$b1, \$_c,

✓ Hằng

- Hằng cũng là một biến nhưng giá trị của nó không thể thay đổi. Tuy nhiên cách khai báo biến và hằng số thì lại khác nhau.
- **Cú Pháp:** `define('ten_hang', 'gia_tri');`

Ví dụ: Tạo một hằng số có tên là SDT và gán giá trị cho nó là 0909090909

```
define('SDT', '0909090909');
```

✓ Các toán tử và kiểu dữ liệu

- **Toán tử gán**: gán giá trị cho 1 biến,
Ví dụ: `$number = 10 ; // gán giá trị 10 cho biến $number`
- **Toán tử số học**: các phép toán cơ bản cộng, trừ, nhân, chia, ...

Tên	Ký hiệu	Mô tả	Ví dụ
Phép cộng	+	Cộng hai số hạng	$\$a + \b
Phép trừ	-	Trừ hai số hạng	$\$a - \b
Phép nhân	*	Nhân hai số hạng	$\$a * \b
Phép chia	/	Chia hai số hạng	$\$a / \b
Phép chia lấy dư	%	Chi lấy dư	$5 \% 2 = 1$

Ví dụ :

```
$a = 5;
$b = 10;
$c = $a + $b;
$d = $a * $b;
```

- **Toán tử so sánh**: các phép toán logic như so sánh bằng, lớn hơn, nhỏ hơn,...

Tên	Ký hiệu	Mô tả	Ví dụ
So sánh bằng	==	Hai số hạng bằng nhau	$\$a == \b
So sánh khác	!=	Hai số hạng khác nhau	$\$a != \b
So sánh lớn hơn	>	So sánh lớn hơn	$\$a > \b
So sánh nhỏ hơn	<	So sánh nhỏ hơn	$\$a < \b
Lớn hơn hoặc bằng	>=	Lớn hơn hoặc bằng	$\$a >= \b
Nhỏ hơn hoặc bằng	<=	Nhỏ hơn hoặc bằng	$\$a <= \b

Ví dụ :

```
$a = 5;
$b = 10;
if ($a > $b) {
    echo "$a lớn hơn $b";
} else {
    echo "$a nhỏ hơn $b";
}
```

- **Toán tử logic**: Toán tử logic là tổ hợp các giá trị boolean có kết quả trả về là TRUE hoặc FALSE

Tên	Ký hiệu	Mô tả	Ví dụ
Phép And	&&	Cả 2 vế phải thoả mãn điều kiện	$\$a > \$b \&\& \$a > \c
Phép Or		Một trong 2 vế thoả mãn điều kiện là được	$\$a > \$b \$a < \c
Phủ định	!	Phủ định 1 điều kiện, giá trị nào đó	$\$a != \b

Ví dụ :

```
$diem = 7;
if($diem <= 5) {
    echo "Bạn học dốt quá!";
}elseif($diem > 5 && $diem <= 8) {
    echo "Bạn học rất khá!";
}else{
    echo "Bạn học rất giỏi";
}
```

- **PHP hỗ trợ 4 kiểu dữ liệu**

- Kiểu số

+ Số nguyên từ -2^{31} đến $2^{31}-1$

❖ Hệ thập phân: \$a = 16;

❖ Hệ 16 (hexa): \$a=0x10;

❖ Hệ 8 (bát phân): 020;

+ Số thực (thập phân): từ $1.7E-308$ đến $1.7E+308$

❖ Biểu diễn: \$a=0.017;

❖ Dạng khoa học: \$a=17.0E-03;

- Kiểu chuỗi

+ Giới hạn bởi nháy đơn (') hoặc kép (")

+ Chuỗi đặt trong nháy kép bị thay thế, trong nháy đơn thì không.

Ví dụ:

```
$a= "Hello";
```

```
$b= "$a world"; //tương đương $b="Hello world"
```

```
$c= ' $a world'; //$c=' $a world'(không thay đổi)
```

- Kiểu logic

+ Có 2 trạng thái: true và false

Ví dụ:

```
$x = true;
```

```
$y = false;
```

1.4. Các cấu trúc điều khiển.

1.4.1.Cấu trúc rẽ nhánh

Dạng 1

```
if (<điều kiện>)
```

```
<công việc>;
```

Mô tả hoạt động

Nếu <điều kiện> là đúng thì thực hiện <công việc>, nếu <điều kiện> là sai thì sẽ không thực hiện <công việc>.

Lưu ý: Nếu <công việc> có từ hai câu lệnh đơn trở lên thì cần đặt <công việc> giữa hai dấu { }.

Dạng 2

```
if (<điều kiện>)
```

```
<công việc 1>;  
else  
    <công việc 2>;
```

Mô tả hoạt động

Nếu <điều kiện> là đúng thì sẽ thực hiện <công việc 1>, ngược lại nếu <điều kiện> là sai thì <công việc 2> sẽ được thực hiện.

1.4.2.Cấu trúc lặp với số lần lặp biết trước

```
for (<biểu thức 1>; <điều kiện>; <biểu thức 2>)  
    <công việc>;
```

Mô tả hoạt động

Đầu tiên sẽ thực hiện <biểu thức 1>, sau đó kiểm tra <điều kiện>. Nếu <điều kiện> là đúng thì sẽ thực hiện <công việc> rồi đến <biểu thức 2>; sau khi thực hiện xong <biểu thức 2> thì cấu trúc này quay trở lại kiểm tra <điều kiện> và bắt đầu một vòng lặp mới,...cứ thế vậy đến khi <điều kiện> nhận giá trị sai thì cấu trúc lặp này sẽ kết thúc.

1.4.3.Sơ đồ lặp với số lần lặp không biết trước**Dạng 1**

```
while (<điều kiện>)  
    <công việc>;
```

Mô tả hoạt động

Nếu <điều kiện> là đúng thì sẽ thực hiện <công việc>, sau đó quay trở lại thực hiện <điều kiện>. Vòng lặp while này sẽ kết thúc khi <điều kiện> là sai.

Dạng 2

```
do  
{  
    <công việc>;  
}  
while (<điều kiện>;)
```

Mô tả hoạt động

Trước hết <công việc> sẽ được thực hiện, sau đó kiểm tra <điều kiện>. Nếu <điều kiện> là đúng thì quay trở lại thực hiện <công việc> và bắt đầu một vòng lặp mới. Vòng lặp do while này sẽ kết thúc khi <điều kiện> nhận giá trị sai.

1.5.Hàm

✓ Khai báo hàm

- `function tên_hàm(ts1, ts2,..., tsn){`
 //các lệnh trong thân hàm
}
- Để trả ra giá trị cho hàm ta sử dụng lệnh **return** biểu_thức;
- Để thoát khỏi hàm: Sử dụng `return` hoặc `exit`
- Ví dụ: Xây dựng hàm tính tổng 2 số

```
function tinh tong($a, $b)
{
    return $a + $b;
}
```

✓ Gọi hàm

- `tên_hàm(gt1, gt2,..., gtn);`

Ví dụ: gọi hàm tính tổng 2 số

```
$so1 = 12;
$so2 = 13;
echo tinh tong($so1, $so2);
```

- **Truyền tham biến và tham trị.**

Việc gọi hàm tính tổng (`tinh tong($so1,$so2)`), tất cả việc truyền biến đều là tham trị. Để truyền tham biến (biến được truyền có thể thay đổi nội dung khi ra khỏi hàm), khi gọi hàm, ta đặt thêm dấu `&` trước biến cần truyền tham biến.

- Ví dụ về truyền tham biến

<?php

```
function F($a, $b){
    $a = 5;
    $b = 5;
    echo "<br> Trong hàm F(), a = ".$a." và b = ".$b;
}

$a = 8;
$b = 8;
echo "<br> 1. Ngoài hàm F(), a = ".$a." và b = ".$b;

//Gọi hàm tham trị.
F($a, $b);
echo "<br>2. Ngoài hàm F(), a = ".$a." và b = ".$b;

//Gọi hàm với biến $a là tham biến.
```

```
F(&$a, $b);
echo "<br>3. Ngoài hàm F()a = ".$a." và b = ".$b;
?>
```

Kết quả:

1. Ngoài hàm F(), a = 8 và b = 8
Trong hàm F(), a = 5 và b = 5
2. Ngoài hàm F(), a = 8 và b = 8
Trong hàm F(), a = 5 và b = 5
3. Ngoài hàm F(), a = 5 và b = 8

1.6 Mảng và chuỗi

- + Khởi tạo mảng một chiều với dữ liệu là chuỗi bằng hàm array()
 - ❖ Chỉ số mặc định với chỉ số đầu bằng 0


```
$a=array("xin", "chào", "bạn");
$a[0]="xin";
$a[1]="chào";
$a[2]="bạn";
```
 - ❖ Chỉ số mặc định với chỉ số đầu khác 0


```
$a=array(5=>"xin", "chào", "bạn");
$a[5]="xin";
$a[6]="chào";
$a[7]="bạn";
```
- + Khởi tạo mảng nhiều chiều với dữ liệu là chuỗi bằng hàm array()


```
$monhoc = array(array("HTML4","HTML5"),
                  array("CSS2","CSS3"),
                  array("Jquery","Jquery Mobile"));
$monhoc[0][0]= "HTML4"
$monhoc[0][1]= "HTML5"
$monhoc[1][0]= "CSS2"
$monhoc[1][1]= "CSS3"
$monhoc[2][0]= "Jquery"
$monhoc[2][1]= "Jquery Mobile"
```

1.7.Tập tin

- ✓ File được tải lên và lưu trên máy chủ có tên là \$_FILES["UserFileName"] với các thông tin như sau:
 - \$_FILES["UserFileName"]['name']: Tên gốc của file trên máy khách
 - \$_FILES["UserFileName"]['type']: Kiểu MIME của file. Ví dụ: "image/gif", "image/jpeg",.....
 - \$_FILES["UserFileName"]['size']: Kích thước của file được tải tính theo byte
 - \$_FILES["UserFileName"]['tmp_name']: Tên tạm của file khi nó đã được tải và lưu trên thư mục tạm của máy chủ.

Để **upload được file**, chúng ta cần thêm thuộc tính **enctype="multipart/form-data"** cho form, ta dùng thẻ input type="file" để **upload file**

- Tạo file form.php với nội dung như sau

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Upload file trong PHP!</title>
</head>

<body>
<h3>Upload file !</h3>
<form name="formUpload" method="post" action="<?php echo
$_SERVER['PHP_SELF'];?>" enctype="multipart/form-data">
Select file:<input type="file" name="file"/>
<input type="submit" name="OK" value="Upload"/>
</form>
</body>
</html>
```

- Tiếp theo ta sẽ kiểm tra xem người dùng đã nhấn nút submit hay chưa bằng cách kiểm tra sự tồn tại của biến \$_POST['ok'], nếu người dùng đã nhấn submit rồi thì tiến hành kiểm tra xem họ đã chọn file hay chưa, nếu chưa chọn file thì báo lỗi

- Ta có thể lấy được tên file, kiểu file, kích cỡ file thông qua các biến :

```
Tên file : $_FILES['file']['name'];
Kiểu file : $_FILES['file']['type'];
Kích thước file : $_FILES['file']['size'];
<?php
    if(isset($_POST['ok'])){ // Người dùng đã ấn submit
        if($_FILES['file']['name'] != NULL){ // Đã chọn file
            // thực hiện công việc upload
        }else{
            echo "Vui lòng chọn file";
        }
    }
?>
```

- Khi người dùng đã chọn file thì tiến hành upload ? tùy theo yêu cầu của bài toán mà chúng ta tiến hành xây dựng chức năng cho nó, ví dụ yêu cầu đặt ra ở đây là chỉ được upload các file ảnh có kiểu file là jpg, png, gif và cỡ file không được lớn hơn 1mb

```
<?php
    if ( $_FILES['file']['type'] == "image/jpeg"
    || $_FILES['file']['type'] == "image/png"
    || $_FILES['file']['type'] == "image/gif"){// là file ảnh
        // Tiến hành code upload
    }else{
        // không phải file ảnh
```

```

        echo "Kiểu file không hợp lệ";
    }
?>

```

- Bước cuối cùng là xem file vừa chọn kích cỡ có nhỏ hơn 1mb không, file size ở đây được tính theo byte (1mb = 1*1024 kb * 1024 bytes) . Nếu cỡ file nhỏ hơn 1mb tiến hành upload file bằng hàm move_uploaded_file , file upload sẽ được lưu trong thư mục images

```

<?php
if($_FILES['file']['size'] > 1048576){
    echo "File không được lớn hơn 1mb";
}else{
    // file hợp lệ, tiến hành upload
    $path= "images/"; // file lưu vào thư mục images
    $tmp_name= $_FILES['file']['tmp_name'];
    $name= $_FILES['file']['name'];
    $type= $_FILES['file']['type'];
    $size= $_FILES['file']['size'];
    // Upload file
    move_uploaded_file($tmp_name,$path.$name);
    echo "File uploaded! <br />";
    echo "Tên file : ".$name."<br />";
    echo "Kiểu file : ".$type."<br />";
    echo "File size : ".$size;
}
?>

```

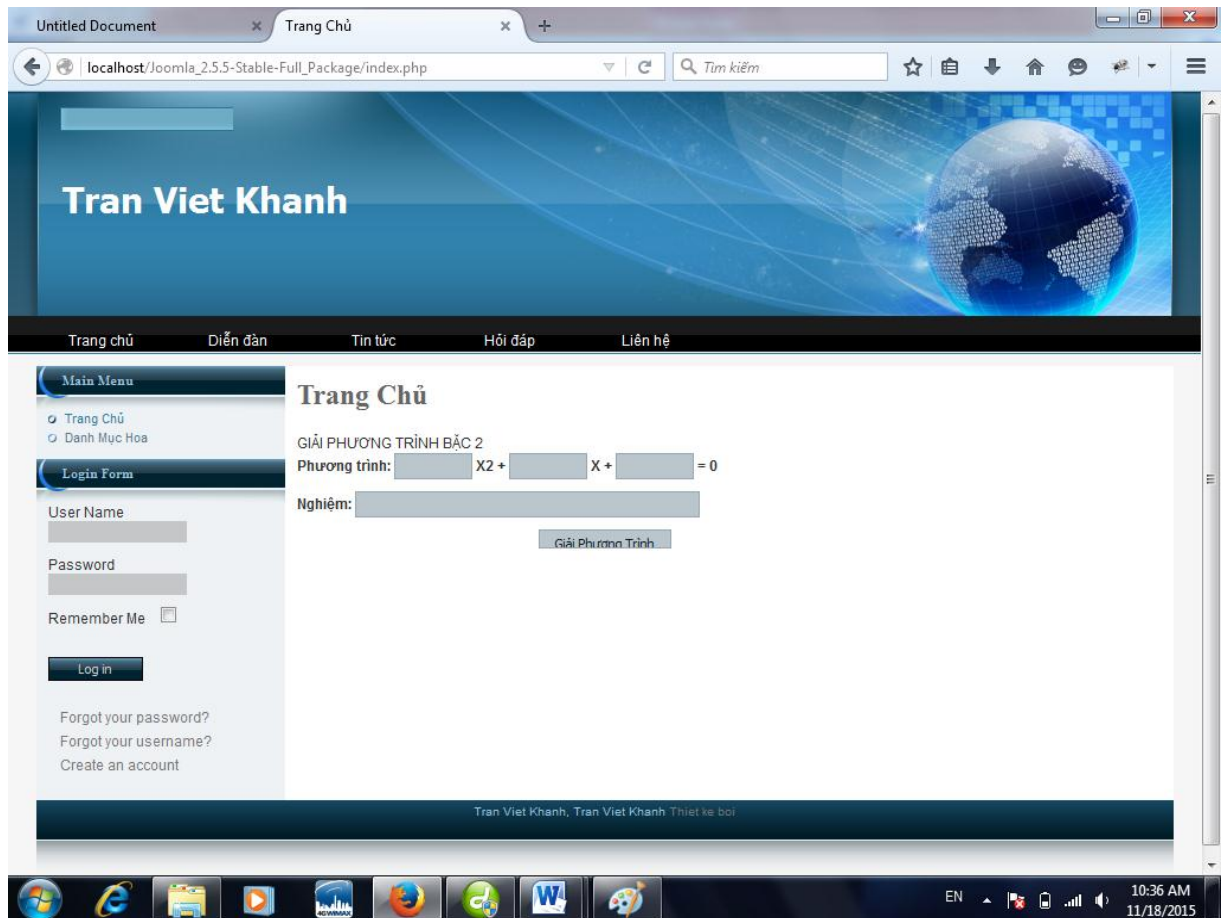
1.8.Tạo trang PHP đầu tiên.

Trong học phần lập trình PHP này, khi viết xong một chương trình ứng dụng web nào đó thì sinh viên cần kiểm thử lại chương trình qua một số bộ test mẫu; để từ đó có thể phát hiện sai sót hoặc có thể cải tiến để được một chương trình hiệu quả hơn về mặt thời gian hoặc không gian lưu trữ dữ liệu,...

B.CÁC DẠNG BÀI TẬP

❖ Dạng 1: Cấu trúc rẽ nhánh

Bài tập 1-1. Xây dựng ứng dụng Web giải phương trình dạng $ax^2 + bx + c = 0$ như hình sau đây:



Hướng dẫn giải bài tập 1-1: Sinh viên thiết kế form giao diện và viết chương trình theo code mẫu sau đây:

<?php

```
$nghiem=NULL;
function GiaiPhuongTrinhBacHai($a,$b,$c)
{
    if($a==0)
    {
        if($b==0)
        {
            if($c==0)
                $nghiem = "Phương trình có vô số nghiệm";
            if($c!=0)
                $nghiem = "Phương trình vô nghiệm";
        }
        else if($b!=0)
        {
            $x=(-($c/$b));
            $x= round($x,2);
            $nghiem = "Phương trình có nghiệm: x = $x";
        }
    }
    if($a!=0)
    {
        $delta = pow($b,2)- 4*$a*$c;
        if($delta<0)
```

```

        $nghiem = "Phương trình vô nghiệm";
    if($delta==0)
    {
        $nghiemkep = (-$b/2*$a);
        $nghiem = "Phương trình có nghiệm số kép x1 = x2 = $nghiemkep";
    }
    if($delta>0)
    {
        @$x1 = (-$b + sqrt($delta))/(2*$a);
        @$x2 = (-$b - sqrt($delta))/(2*$a);
        $nghiem = "Phương trình có hai nghiệm phân biệt x1=".round($x1,2).",
                                                    x2=".round($x2,2);
    }
    }
    return $nghiem;
}
if(isset($_POST["a"]) && isset($_POST["b"]) && isset($_POST["c"]))
{
    $a=$_POST["a"];
    $b=$_POST["b"];
    $c=$_POST["c"];
    $nghiem=GiaiPhuongTrinhBacHai($a,$b,$c);
}
?>
<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="post">
    <fieldset><legend>GIẢI PHƯƠNG TRÌNH BẬC 2</legend>
    <p><b>Phương trình:</b>
        <input type="text" name="a" size="5" maxlength="5" value="<?php if(isset($_POST['a']))
        echo $_POST['a'];?>" />
        <b> X2 +</b>
        <input type="text" name="b" size="5" maxlength="5" value="<?php if(isset($_POST['b']))
        echo $_POST['b'];?>" />
        <b> X +</b>
        <input type="text" name="c" size="5" maxlength="5" value="<?php if(isset($_POST['c']))
        echo $_POST['c'];?>" />
        <b> = 0</b>
    </p>
    <p><b>Nghiệm:</b>
        <input type="text" name="nghiem" size="50" maxlength="50" value="<?php echo
        $nghiem;?>" />
    </p>
</fieldset>
<div align="center"><input type="submit" name="submit" value="Giải Phương Trình" /> </div>
</form>

```

➤ **Một số kết quả khi chạy chương trình:**

Số lần test	Nhập số a	Nhập số b	Nhập số c	Kết quả
test 1	0	0	0	Phương trình vô nghiệm
test 2	0	0	1	Phương trình vô nghiệm
test 3	0	2	5	Phương trình có một nghiệm x=-2.5
test 4	1	3	4	Phương trình vô nghiệm
test 5	1	-4	4	Phương trình có nghiệm kép $x_1=x_2=2$

test 6	1	4	3	Phương trình có hai nghiệm $x_1=-1, x_2=-3$
--------	---	---	---	---

❖ Dạng 2: Cấu trúc lặp với số lần lặp biết trước

Bài tập 1-2. Xây dựng ứng dụng Web tính tổng $S(n) = 1 + 2 + \dots + n$ với n là số tự nhiên ≥ 0 như hình sau đây:



Hướng dẫn giải bài tập 1-2: Sinh viên thiết kế form giao diện và viết chương trình theo code mẫu sau đây:

<?php

```
error_reporting("E_ERROR");
$stong=NULL;
function Sn($n)
{
    for($i=0;$i<=$n;$i++)
        $stong=$stong+$i;
    return $stong;
}
if(isset($_POST["n"]))
{
    $n=$_POST["n"];
    $stong=Sn($n);
```

}

?>

```

<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="post">
  <fieldset><legend><font color="#FF0000" size="+1">Tính tổng  $S(n) = 1 + 2 + \dots + n$  với  $n$  là số tự
  nhiên  $\geq 0$ </font></legend>
  <p><b>Nhập n:</b><input type="text" name="n" size="5" maxlength="5" value="<?php
  if(isset($_POST['n'])) echo $_POST['n'];?>" /></p>
  <p><b>S(n):</b>
  <input type="text" name="tong" size="50" maxlength="50" value="<?php echo $tong;?>" /></p>
</fieldset>
<div align="center"><input type="submit" name="submit" value="Tính tổng" /> </div>
</form>

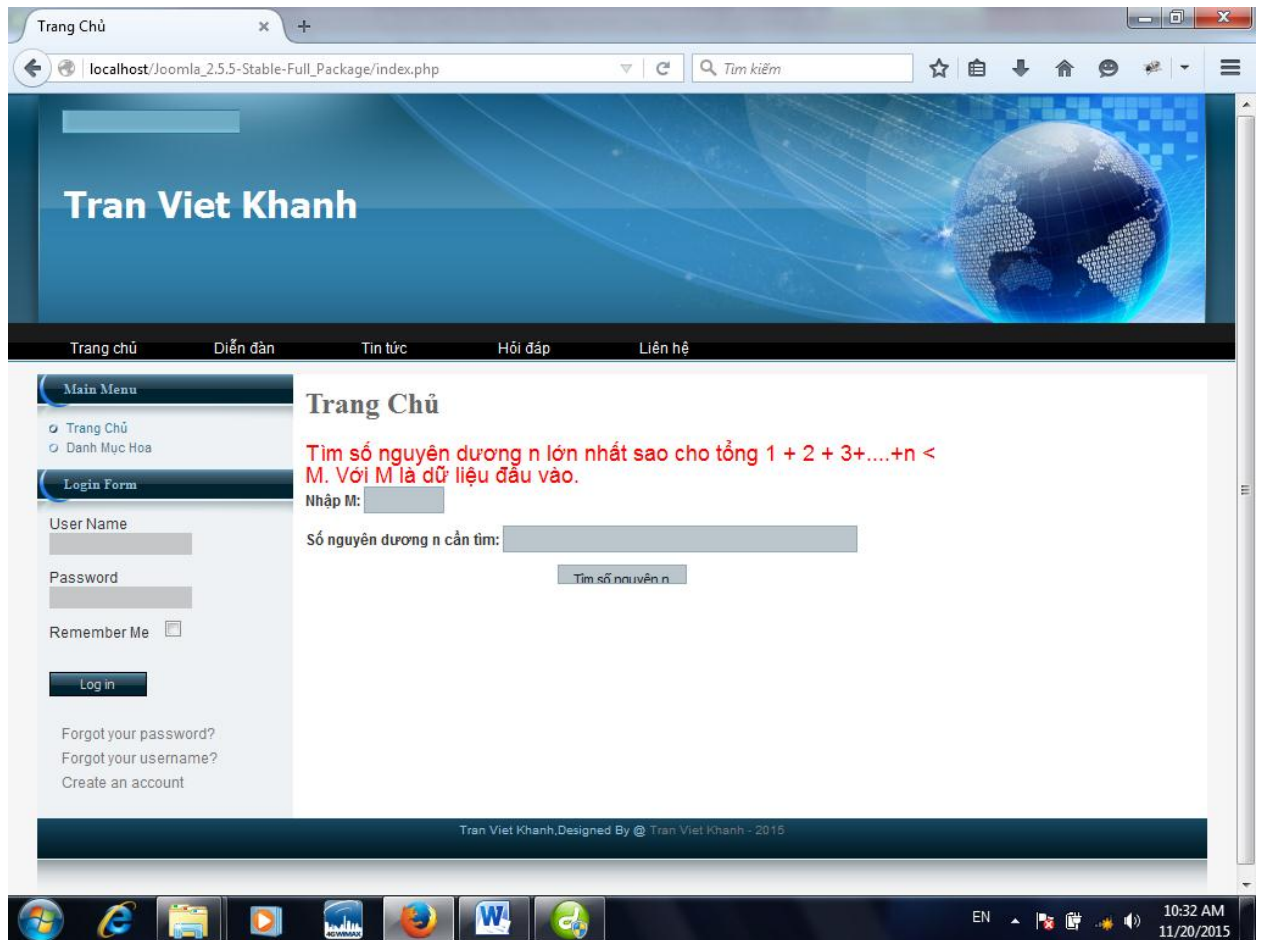
```

➤ **Một số kết quả khi chạy chương trình:**

Số lần test	Nhập N	S(N)
test 1	0	0
test 2	1	1
test 3	10	55
test 4	100	5050

❖ **Dạng 3: Cấu trúc lặp với số lần lặp không biết trước**

Bài tập 1-3. Xây dựng ứng dụng Web tìm số nguyên dương n lớn nhất sao cho tổng $1 + 2 + 3 + \dots + n < M$ với M là dữ liệu đầu vào như hình sau đây:



Hướng dẫn giải bài tập 1-3: Sinh viên thiết kế form giao diện và viết chương trình theo code mẫu sau đây:

+ Sử dụng vòng lặp while

<?php

```
$n=NULL;
function MaxN($M)
{
    $n=0;
    $s=0;
    while ($s<$M)
    {
        $n++;
        $s=$s+$n;
    }
    return $n-1;
}
if(isset($_POST["M"]))
{
    $M=$_POST["M"];
    $n=MaxN($M);
}
```

?>

```
<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="post">
  <fieldset><legend><font color="#FF0000" size="+1">Tìm số nguyên dương n lớn nhất
  sao cho tổng 1 + 2 + 3 + ... + n < M. Với M là dữ liệu đầu vào.</font></legend>
```

```

<p><b>Nhập M:</b>
<input type="text" name="M" size="5" maxlength="5" value="<?php
if(isset($_POST['M'])) echo $_POST['M'];?>" />
</p>
<p><b>Số nguyên dương n cần tìm:</b>
<input type="text" name="n" size="50" maxlength="50" value="<?php echo $n;?>" />
</p>
</fieldset>
<div align="center"><input type="submit" name="submit" value="Tìm số nguyên n" />
</div>
</form>
+ Sử dụng vòng lặp do...while

```

```

<?php
    $n=NULL;
    function MaxN($M)
    {
        $n=0;
        $s=0;
        do
        {
            $n++;
            $s=$s+$n;
        }
        while($s<$M);
        return $n-1;
    }
    if(isset($_POST["M"]))
    {
        $M=$_POST["M"];
        $n=MaxN($M);
    }
    ?>

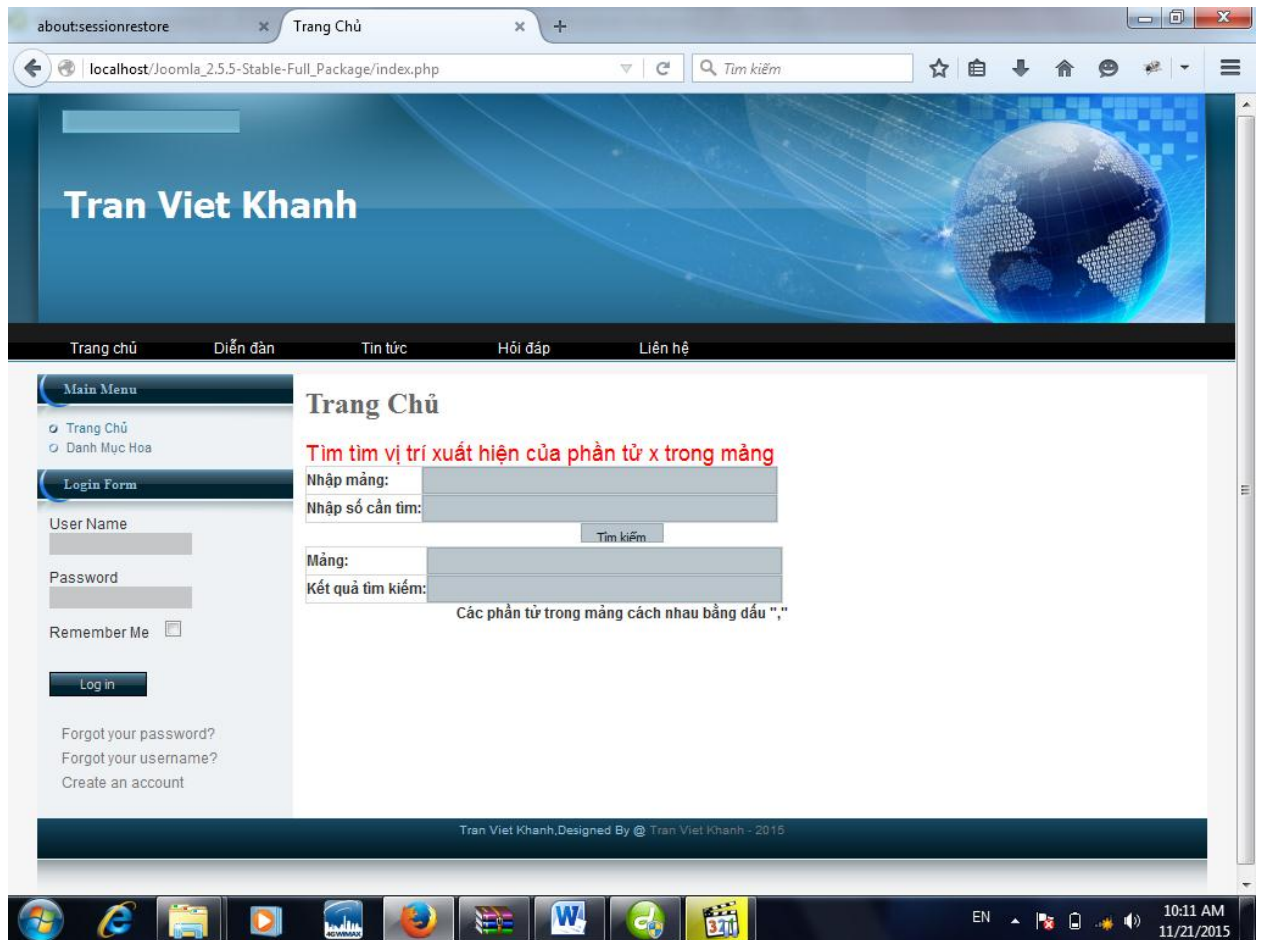
```

➤ **Một số kết quả khi chạy chương trình:**

Số lần test	test 1	test 2	test 3	test 4
M	15	20	100	1000
N	4	5	13	44

❖ Mảng và chuỗi

Bài tập 1-4. Xây dựng ứng dụng Web tìm vị trí xuất hiện của phần tử x trong mảng như hình sau đây:



Hướng dẫn giải bài tập 1-4: Sinh viên thiết kế form giao diện và viết chương trình theo code mẫu sau đây:

<?php

```
$n=NULL;
function TimKiem($mang,$giatri)
{
    $n=count($mang);
    $kq=-1;
    for($i=0;$i<$n;$i++)
    {
        if($mang[$i]==$giatri)
        {
            $kq=$i;
            break;
        }
    }
    return $kq;
}
if(isset($_POST["mang"])&&isset($_POST["x"]))
{
    $mangchuoi=trim($_POST["mang"]);
    $giatri=$_POST["x"];
    $mang = explode(",", $mangchuoi);
```

```

    $kq = TimKiem($mang,$giatri);
    $mangketqua = implode(",",$mang);
    if($kq!=-1)
    {
        $kq=$kq+1;
        $chuoi = "Tìm thấy số $giatri tại vị trí $kq";
    }
    else
        $chuoi = "Không tìm thấy số $giatri trong mảng";
}

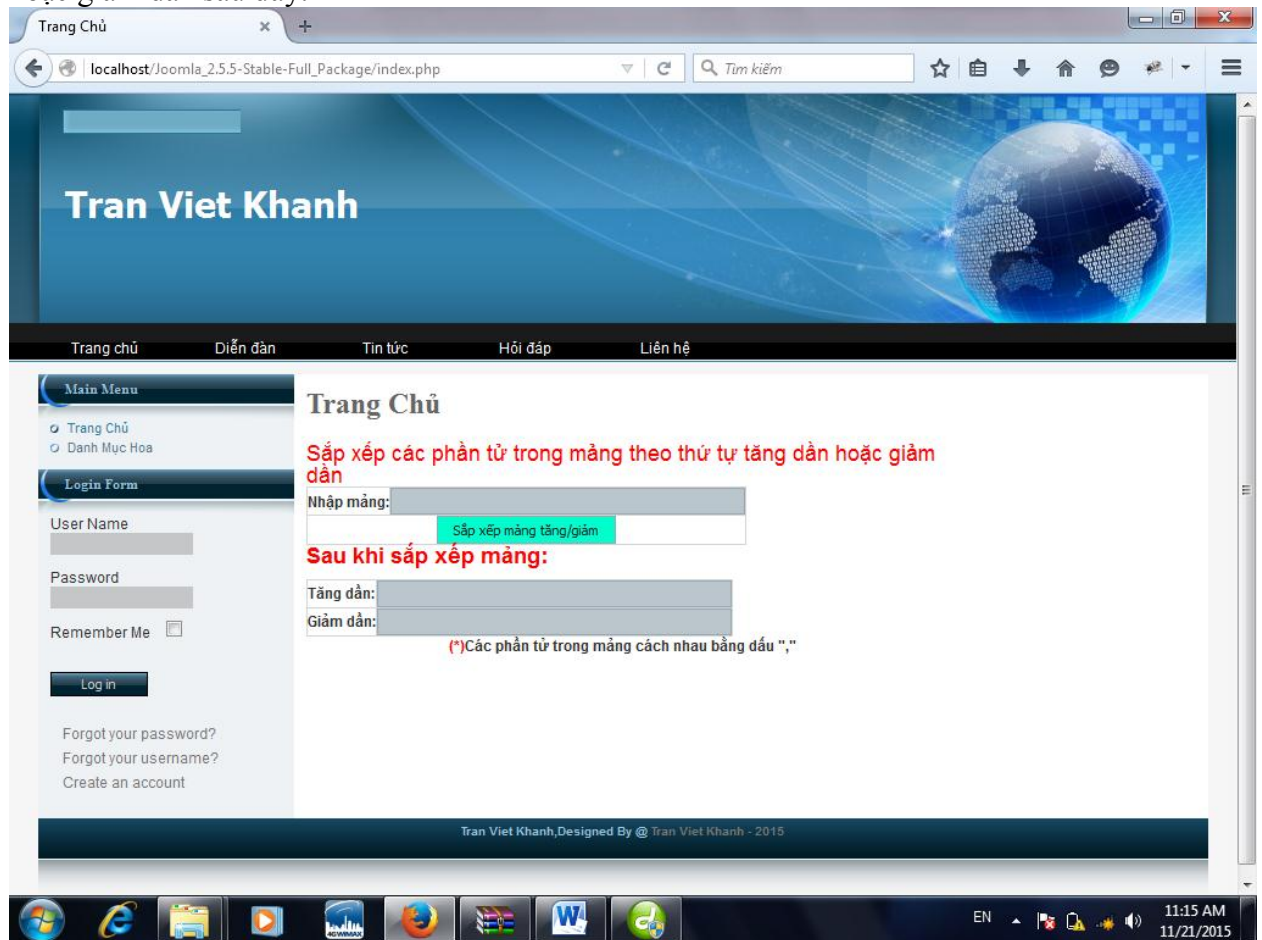
?>
<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="post">
    <fieldset><legend><font color="#FF0000" size="+1">Tìm tìm vị trí xuất hiện của
    phần tử x trong mảng</font></legend>
    <table border="1">
        <tr>
            <td><b>Nhập mảng:</b></td>
            <td><input type="text" name="mang" size="50" maxlength="50"
            value="<?php if(isset($_POST['mang'])) echo $_POST['mang']; ?>" /></td>
        </tr>
        <tr>
            <td><b>Nhập số cần tìm:</b></td>
            <td><input type="text" name="x" size="50" maxlength="50" value="<?php
            if(isset($_POST['x'])) echo $_POST['x'];?>" /></td>
        </tr>
    </table>
    <div align="center"><input type="submit" name="submit" value="Tìm kiếm"/></div>
    <table border="1">
        <tr>
            <td><b>Mảng:</b></td>
            <td><input type="text" name="xuatmang" size="50" maxlength="50"
            value="<?php echo $mangketqua;?>" /></td>
        </tr>
        <tr>
            <td><b>Kết quả tìm kiếm:</b></td>
            <td><input type="text" name="vt" size="50" maxlength="50" value="<?php
            echo $chuoi;?>" /></td>
        </tr>
    </table>
    <div align="center"><b>Các phần tử trong mảng cách nhau bằng dấu "," </b></div>
</fieldset>
</form>

```

➤ **Một số kết quả khi chạy chương trình:**

Số lần test	Mảng	X	Vị trí xuất hiện
test 1	4,5,3,2,7,8	2	4
test 2	1,2,4,3,9,8,5,6	7	Không tìm thấy
test 3	1,2,4,3,9,8,5,6	9	5

Bài tập 1-5. Xây dựng ứng dụng Web để sắp xếp các phần tử trong mảng theo thứ tự tăng dần hoặc giảm dần sau đây:



Hướng dẫn giải bài tập 1-5: Sinh viên thiết kế form giao diện và viết chương trình theo code mẫu sau đây:

<?php

```
$n=NULL;
function HoanVi(&$a,&$b)
{
    $t=$a;
    $a=$b;
    $b=$t;
}
function SapTang($mang)
{
    $n=count($mang);
    for($i=0;$i<$n-1;$i++)
        for($j=$i+1;$j<$n;$j++)
            if($mang[$i]>$mang[$j])
                HoanVi($mang[$i],$mang[$j]);
    return $mang;
}
function SapGiam($mang)
{
    $n=count($mang);
    for($i=0;$i<$n-1;$i++)
```

```

        for($j=$i+1;$j<$n;$j++)
            if($mang[$i]<$mang[$j])
                HoanVi($mang[$i],$mang[$j]);
    return $mang;
}
if(isset($_POST["mang"]))
{
    $mangchuoi=trim($_POST["mang"]);
    $mang = explode(",", $mangchuoi);

    $mang1 = SapTang($mang);
    $chuoitang = implode(",", $mang1);

    $mang2 = SapGiam($mang);
    $chuoigiam = implode(",", $mang2);

}
?>
<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="post">
    <fieldset><legend><font color="#FF0000" size="+1">Sắp xếp các phần tử trong mảng theo
    thứ tự tăng dần hoặc giảm dần</font></legend>
        <table border="1">
            <tr>
                <td><b>Nhập mảng:</b></td>
                <td><input type="text" name="mang" size="50" maxlength="50" value="<?php
                if(isset($_POST['mang'])) echo $_POST['mang'];?>" /></td>
            </tr>
            <tr>
                <td colspan="2" style="text-align: center;">
                    <input type="submit"
                    style="background-color:#00FFCC; height:inherit" name="submit" value="Sắp xếp
                    mảng tăng/giảm"/>
                </td>
            </tr>
        </table>
        <p><b><font color="#FF0000" size="+1">Sau khi sắp xếp mảng:</font><b></p>
        <table border="1">
            <tr>
                <td><b>Tăng dần:</b></td>
                <td><input type="text" name="mangtang" size="50" maxlength="50"
                value="<?php echo $chuoitang;?>" /></td>
            </tr>
            <tr>
                <td><b>Giảm dần:</b></td>
                <td><input type="text" name="manggiảm" size="50" maxlength="50"
                value="<?php echo $chuoigiam;?>" /></td>
            </tr>
        </table>
        <div align="center"><b><font color="#FF0000">(*)</font>Các phần tử trong mảng
        ách nhau bằng dấu "," </b></div>
    </fieldset>
</form>

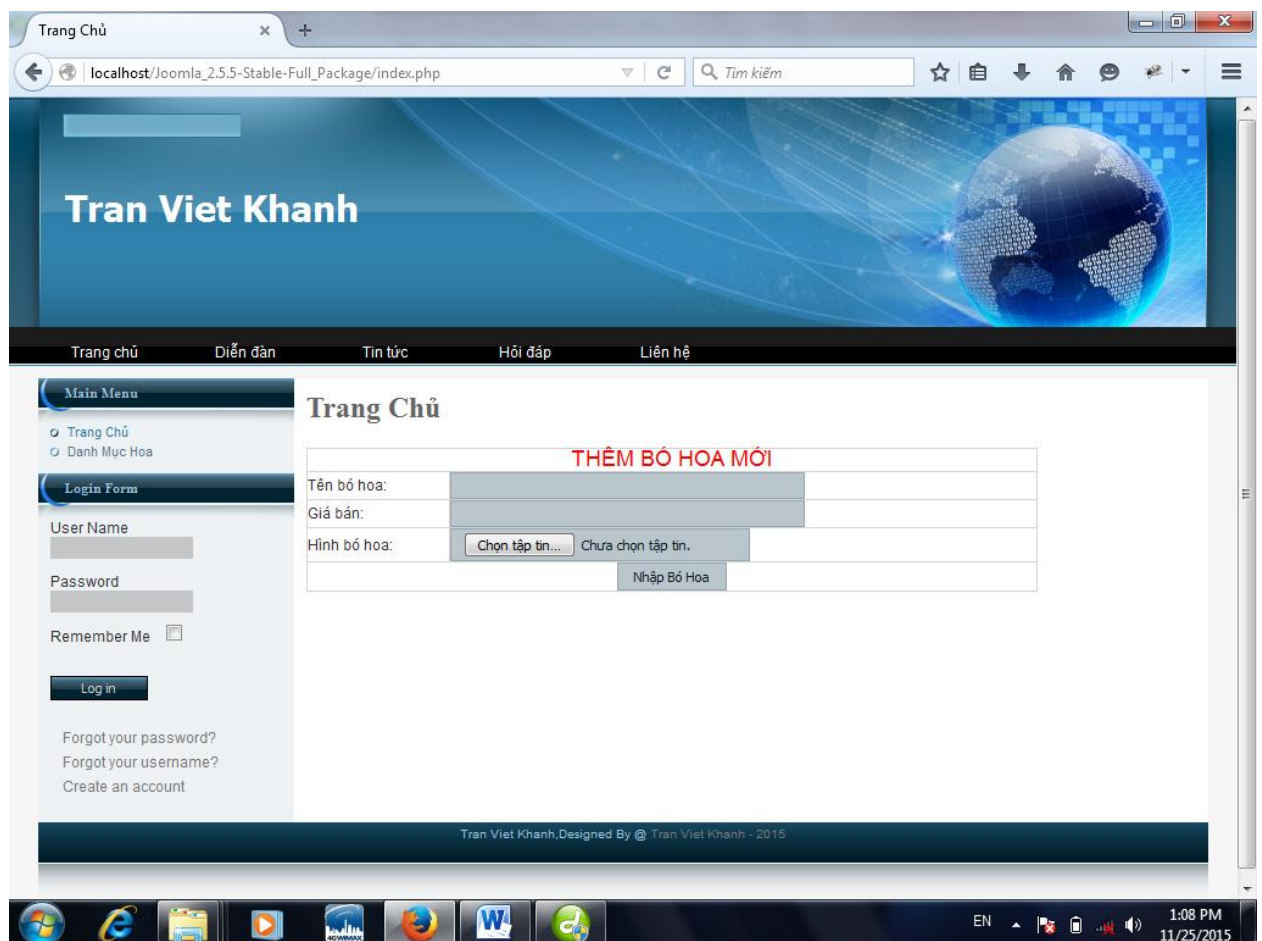
```

➤ Một số kết quả khi chạy chương trình:

Số lần test	Mảng	Mảng tăng	Mảng giảm
test 1	3,1,5,8,9,7,4,6	1,3,4,5,6,7,8,9	9,8,7,6,5,4,3,1
test 2	1,2,4,3,9,8,5,6	1,2,3,4,5,6,8,9	9,8,6,5,4,3,2,1
test 3	5,3,6,7,9,1,2,8,4	1,2,3,4,5,6,7,8,9	9,8,7,6,5,4,3,2,1

❖ Tập tin

Bài tập 1-6. Xây dựng ứng dụng Web nhập thông tin bó hoa và lưu trữ vào file trong thư mục tap_tin, hình ảnh được upload vào thư mục hình_anh:



Hướng dẫn giải bài tập 1-6: Sinh viên thiết kế form giao diện và viết chương trình theo code mẫu sau đây:

```
<?php
```

```
if(isset($_POST["ten_hoa"])&&isset($_POST["gia_ban"]))
{
    if(($_FILES["file"]["type"]=="image/gif")||($_FILES["file"]["type"]=="image/jpeg")
    &&($_FILES["file"]["size"]<60000))
```

```

{
    if($_FILES["file"]["error"]>0)
    {
        echo "Return Code: ".$_FILES["file"]["error"]."<br>";
    }
    else
    {
        move_uploaded_file($_FILES["file"]["tmp_name"],"templates/siteground-
j16-10/hinh_anh/".$_FILES["file"]["name"]);
        $duong_dan="templates/siteground-j16-10/hinh_anh/".$_FILES["file"]
["name"] ;
    }
    $ten_hoa=$_POST["ten_hoa"];
    $gia_ban=$_POST["gia_ban"];
    //$hinh_anh=$_FILES["file"]["name"];
    $noidung="/*".$ten_hoa."|".$gia_ban."|".$duong_dan;
    $file=fopen("tap_tin/hoa_xuan.txt","a",1);
    fwrite($file,$noidung);
    echo "<p align='center'><b>Đã ghi file thành công !</b></p>";
    fclose($file);
    // B2: Xuất bó hoa vừa ghi ra màn hình
    $gia_ban = number_format($gia_ban,0,',',');
    echo "<p align='center'>Kết quả:<br><img src='".$duong_dan'><br>Tên bó
hoa:<b>$ten_hoa</b><br>Giá bán:<b>$gia_ban</b> VNĐ";
}
else
{
    echo "Invalid file";
}
}
else
{
    echo "<p align='center'>Hãy nhập đầy đủ thông tin</p>";
}
}

```

?>

```

<form name="frmNhapBoHoa" method="post" action="<?php echo $_SERVER
[PHP_SELF];?>" enctype="multipart/form-data">
<table width="600" border="1">
    <tr>
        <td colspan="2" align="center"><font color="#FF0000" size="+1">THÊM BÓ HOA
MỚI</font></td>
    </tr>
    <tr>
        <td>Tên bó hoa:</td>
        <td><input type="text" name="ten_hoa" size="50" maxlength="50" /></td>
    </tr>
    <tr>
        <td>Giá bán:</td>
        <td><input type="text" name="gia_ban" size="50" maxlength="50" /></td>
    </tr>

```

```

<tr>
    <td>Hình bó hoa:</td>
    <td><input type="file" name="file" style="height:inherit"/></td>
</tr>
<tr>
    <td colspan="2" align="center"><input type="submit" name="btnNhap" value="Nhập
    Bó Hoa" style="height:inherit"/></td>
</tr>
</table>
</form>

```

➤ **Một số kết quả khi chạy chương trình:**

Số lần test	Thông tin lưu vào file tap_tin/hoa_xuan.txt	Hình được upload vào thư mục hình_anh	Kết quả
test 1	/*Hoa Cúc Lợn 999999 templates/siteground-j16-10/ hinh_anh/hoaculon.gif	hoaculon.gif	Successful
test 2	/* Hoa Cúc Lợn 999999 templates/siteground-j16- 10/hinh_anh/hoaculon.gif/*Hoa Mồng Gà 555555 templates/ siteground-j16-10/hinh_anh/ hoamongga.jpg	hoamongga.jpg	Successful
test 3	Lỗi do kích thước file quá lớn	Lỗi file	Invalid file

C.BÀI TẬP NÂNG CAO

Xây dựng ứng dụng Web cho các bài tập sau:

BT1-7. Cho số tự nhiên n . Kiểm tra xem n có phải là số hoàn chỉnh hay không? (n là số hoàn chỉnh nếu nó bằng tổng các ước số thực sự của nó). Ví dụ 6 là số hoàn chỉnh vì $6=1+2+3$.

Gợi ý làm bài: Thiết kế form giao diện gần giống **BT1-3** bao gồm:

- Textfield cho phép nhập số tự nhiên n .
- Textfield hiển thị kết quả n có là số hoàn chỉnh hay không?
- Lấy giá trị biến n trên form thông qua biến $$_POST$.
- Viết hàm kiểm tra n có là số hoàn chỉnh hay không (kết hợp if và vòng lặp for)
- Nếu n là số hoàn chỉnh hàm cho kết quả là true, ngược lại cho kết quả là false (return true hoặc false)

BT1-8. Cho số tự nhiên n . Kiểm tra xem n có phải là số chính phương hay không?

Ví dụ 1,4,9,16 là các số chính phương.

Gợi ý làm bài: Thiết kế form giao diện gần giống **BT1-3** bao gồm:

- Textfield cho phép nhập số tự nhiên n.
- Textfield hiển thị kết quả n có là số chính phương hay không?
- Lấy giá trị biến n trên form thông qua biến \$_POST.
- Viết hàm kiểm tra n có là số chính phương hay không (dùng if và hàm lấy căn bậc 2 của số nguyên n là **sqrt(n)**)
- Nếu n là số chính phương hàm cho kết quả là true, ngược lại cho kết quả là false (return true hoặc false)

BT1-9. Cho số tự nhiên n. Kiểm tra xem n có phải là số nguyên tố hay không ?

Ví dụ 2,3,5,7 là các số nguyên tố.

Gợi ý làm bài: Thiết kế form giao diện gần giống **BT1-3** bao gồm:

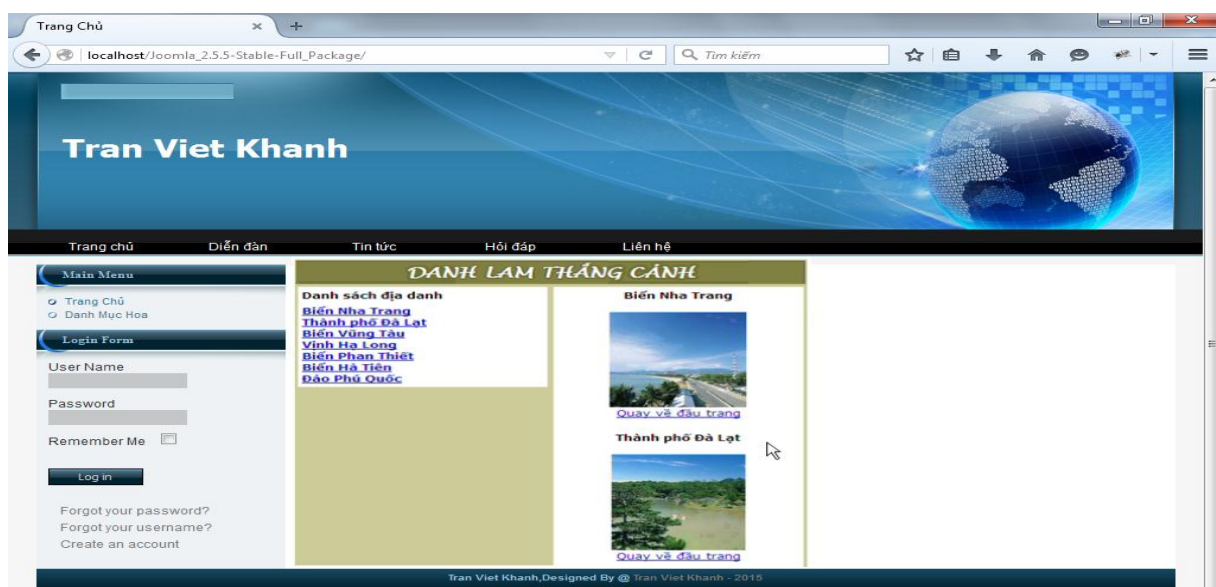
- Textfield cho phép nhập số tự nhiên n.
- Textfield hiển thị kết quả n có là số nguyên tố hay không?
- Lấy giá trị biến n trên form thông qua biến \$_POST.
- Viết hàm kiểm tra n có là số nguyên tố hay không (kết hợp if và vòng lặp for).
- Nếu n là số nguyên tố hàm cho kết quả là true, ngược lại cho kết quả là false (return true hoặc false).

BT1-10. Cho số tự nhiên n. Tìm các số nguyên tố nhỏ hơn hoặc bằng n. Ví dụ: n=8 thì kết quả là: 2,3,5,7

Gợi ý làm bài: Thiết kế form giao diện gần giống **BT1-3** bao gồm:

- Textfield cho phép nhập số tự nhiên n.
- Textfield hiển thị các số nguyên tố tìm được thỏa điều kiện $\leq n$?
- Lấy giá trị biến n trên form thông qua biến \$_POST.
- Viết hàm tìm các số nguyên tố thỏa điều kiện (sử dụng hàm kiểm tra số nguyên tố của **BT1-9** kết hợp if và vòng lặp while hoặc do...while).
- Nếu n là số nguyên tố hàm cho kết quả là true, ngược lại cho kết quả là false (return true hoặc false).

BT1-11. Tạo website danh lam thắng cảnh như hình sau đây



Gợi ý làm bài: Thiết kế form giao diện như hình.

- Tạo ra một mảng hai chiều mặc định chứa danh lam thắng cảnh, mỗi mảng con trong mảng hai chiều gồm có ba phần tử là mã danh lam thắng cảnh, tên danh lam thắng cảnh, và hình ảnh. Ví dụ: (“hl”, ”Vịnh Hạ Long”, ”halong.jpg”). Khi người dùng chưa chọn danh lam thắng cảnh thì hiển thị toàn bộ các danh lam thắng cảnh, khi người dùng chọn danh lam thắng cảnh nào thì sẽ di chuyển tới đúng vị trí danh lam thắng cảnh đó. Khi người dùng chọn quay về đầu trang thì sẽ di chuyển lên đầu trang.
- Thêm vào một “neo” để link có thể dẫn tới “neo” này khi người dùng chọn quay về đầu trang ở mục tiêu đề DANH LAM THẮNG CẢNH.
- Viết code hiển thị tất cả danh lam thắng cảnh ở mục danh sách các địa danh và hiển thị chi tiết từng địa danh khi người dùng chọn trong danh sách các địa danh.

BT1-12. Tạo website nhập thông tin bó hoa như **Bài tập 1-6** cho phép người dùng có thể upload được các loại file ảnh và có kích thước không vượt quá 2MB.

Gợi ý làm bài: Thiết kế form giao diện như **Bài tập 1-6**

- Viết code bổ sung vào biến `$_FILES["file"]["type"]` cho phép người dùng có thể chọn tất cả các loại file ảnh (image/*.*) và `$_FILES["file"]["size"]` có kích thước không vượt quá 2MB.

Chương 2

THIẾT KẾ GIAO DIỆN HTML – CSS

A.TÓM TẮT LÝ THUYẾT

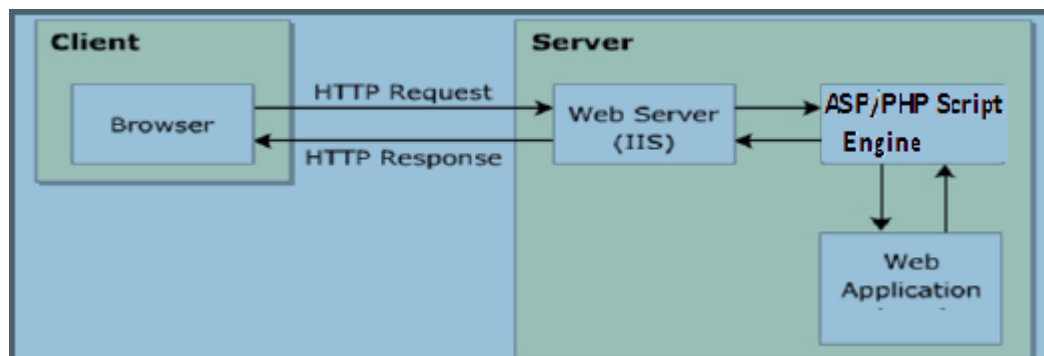
§1: CÁC PHƯƠNG PHÁP GIAO TIẾP GIỮA NGƯỜI DÙNG VÀ MÁY CHỦ

1.1. Web Client - Web Server

1.1.1.Web Client

Dịch vụ World Wide Web (www) được xây dựng theo mô hình Client/Server, tức là người ta thiết lập một máy phục vụ cho việc lưu trữ dữ liệu gọi là Web Server. Phía người sử dụng sẽ có một máy tính có cài phần mềm giao tiếp được với Web Server gọi là Web Browser hay Web Client. Khi người sử dụng gửi URL đến Web Server để yêu cầu sử dụng một tài liệu nào đó thì Web Server sẽ phục vụ tài liệu đó và Web Browser (Web Client) sẽ diễn dịch các tài liệu đó dưới dạng HTML và hiển thị nó lên màn hình với khuôn dạng thích hợp.

Khái niệm URL (Uniform Resource Locator) là một con trỏ được dùng với mục đích là xác định vị trí tài nguyên của môi trường Internet. Thông qua URL mà Web Browser có thể gửi yêu cầu đến Web Server các loại dịch vụ Web khác nhau. Web Browser và Web Server giao tiếp với nhau thông qua giao thức HTTP.



1.1.2.Web Server

Là một phần mềm được cài đặt ở máy Server luôn **lắng nghe** và phục vụ yêu cầu từ Web Browser thông qua một cổng truyền thông nào đó (ngầm định là port 80). Khi máy khách yêu cầu một tài nguyên internet (dịch vụ Web), Web Server sẽ tìm tài nguyên của mình xem có đáp ứng được đòi hỏi đó không, nếu có sẽ gửi kết quả về phía Client.

Web động và Web tĩnh: Ứng dụng Web luôn tồn tại hai loại là trang Web động và tĩnh. Trang Web tĩnh là trang HTML không có kết nối cơ sở dữ liệu. Ngược lại, trang Web động là

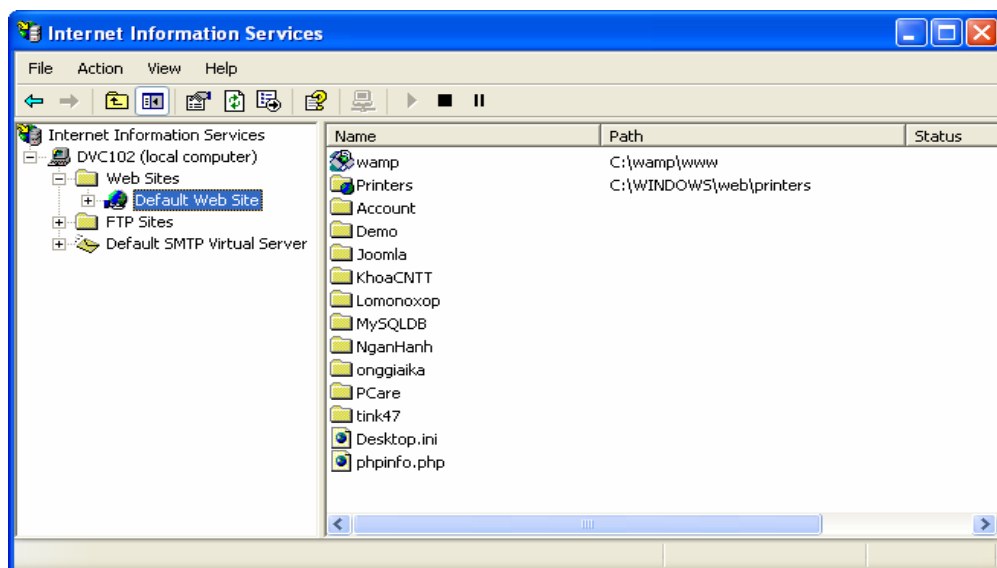
trang Web có kết nối cơ sở dữ liệu. Điều này có nghĩa là mỗi khi trang Web động được làm mới dữ liệu trình bày trên trang Web nó sẽ lấy dữ liệu mới này từ cơ sở dữ liệu.

Trong môi trường Windows, Web Server thường được sử dụng là IIS (Internet Information Services). IIS sử dụng cho các Server Script như: ASP.NET, JSP, PHP, Perl,...

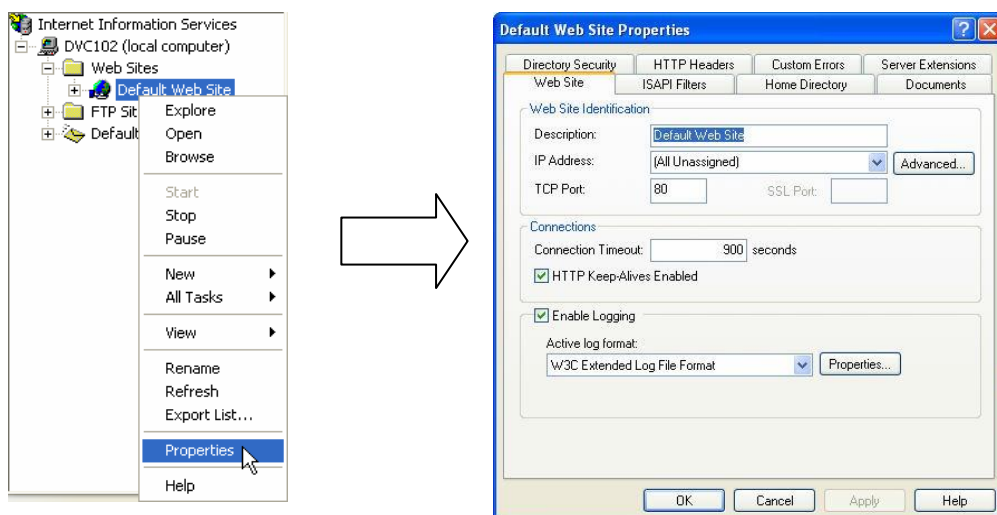
Trong môi trường Linux, Web Server thường sử dụng như Apache, JRun, Web logic,...

1.2. Cấu hình Web Server (IIS)

- ✓ **Giao diện cấu hình:** *Control Panel- > Administrative Tools- > Internet Services Manager*
- ✓ **Web root:** Thư mục gốc của web



- ✓ **Các thông số chung:** *Kích chuột phải vào web cần cấu hình -> Properties*



- ✓ **Một số thông số:**

- Website/ TCP Port : Cổng của dịch vụ web
- Home Directory : Thư mục gốc của web (webroot)
- Documents: Trang mặc định

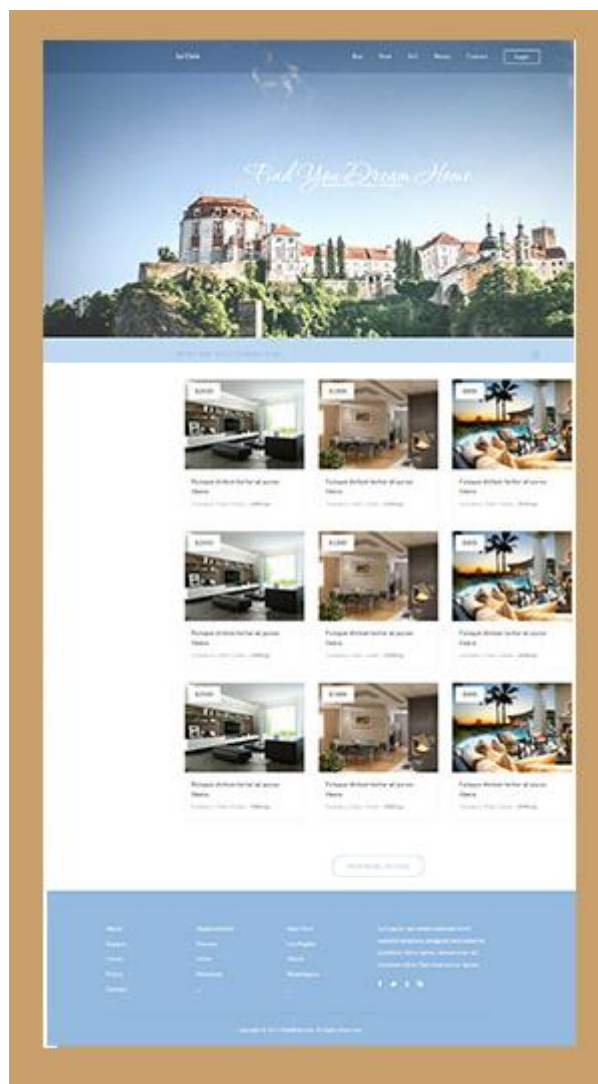
§2: CÁCH VIẾT DỮ LIỆU ĐỘNG TỪ PHP RA GIAO DIỆN HTML - CSS

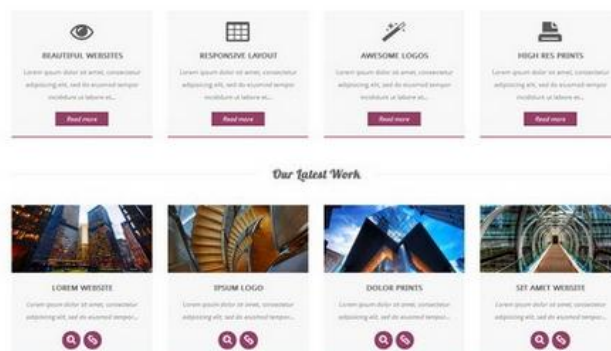
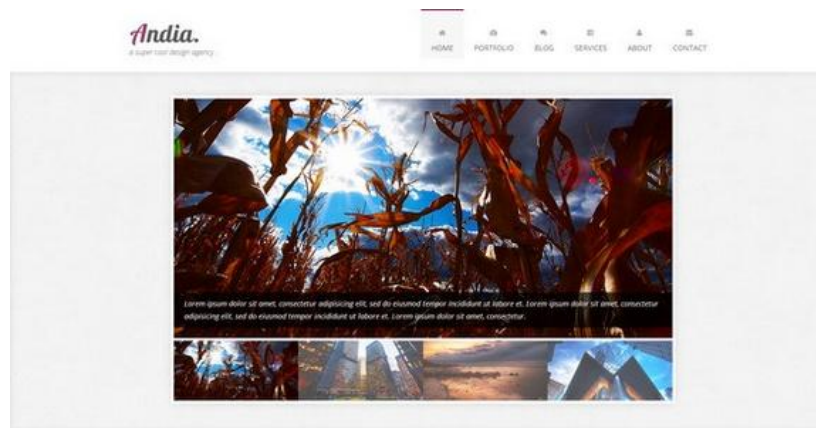
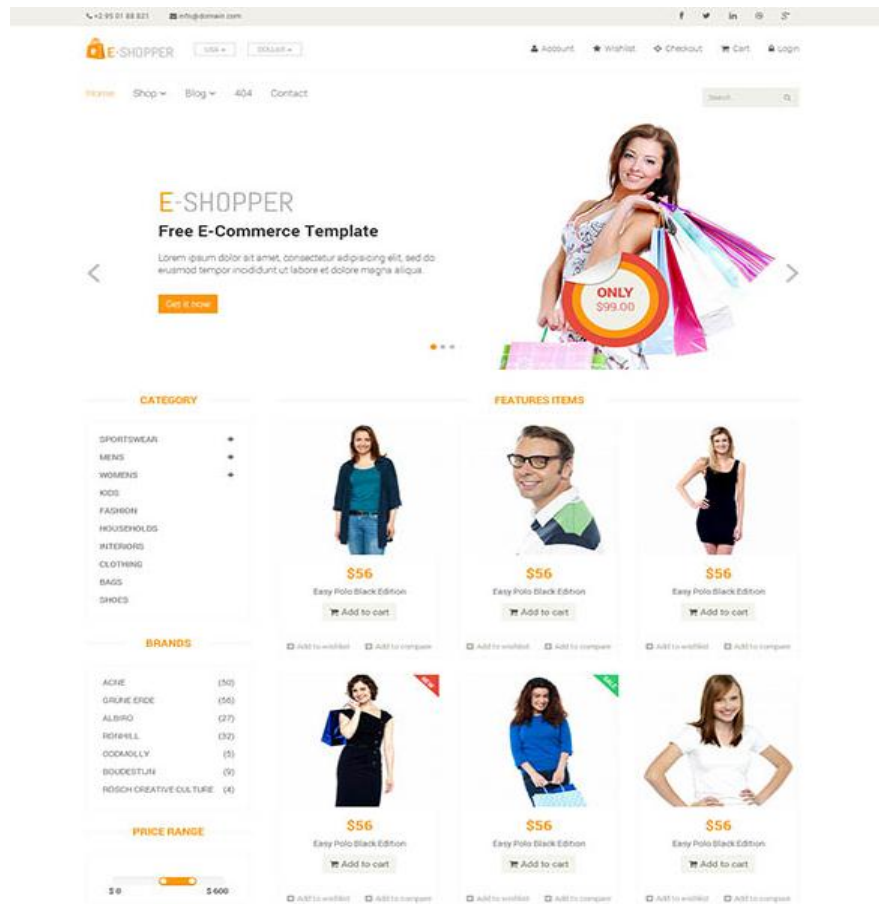
2.1 Tạo biểu mẫu HTML - CSS.

Khác với HTML, HTML5 có thêm một số các tag markup mới sau đây:

- **Tag <header>** và **<footer>** để giúp cô lập các đỉnh và đáy của khối nội dung. Có thể được sử dụng nhiều hơn một lần trên một trang duy nhất.
- **Tag <article>** trong đó xác định cụ thể một phần duy nhất của nội dung, ví dụ như, một bài đăng blog hay lời nhận xét của người dùng.
- **Tag <nav>** để xác định các chỉ tiêu được xem là khối điều hướng.
- **Tag <section>** cho phép bạn xác định một phần chung của nội dung, tương tự như các tag <div> hiện đang tồn tại.
- **Tag <audio>** và **<video>** để đánh dấu sự bao gồm các nội dung audio hay video
- **Tag <canvas>** cho phép vẽ đồ họa sử dụng một ngôn ngữ kịch bản riêng biệt.
- **Tag <embed>** để nhúng nội dung bên ngoài hoặc các ứng dụng vào trang.

Một số biểu mẫu được thiết kế bằng HTML5 – CSS3:





2.2 Nhúng mã PHP vào biểu mẫu HTML.

- + Chúng ta có thể nhúng mã PHP vào mọi vị trí trong biểu mẫu HTML
- + Nhúng mã PHP ngoài thẻ form
- + Nhúng mã PHP bên trong thẻ form

2.3 Gửi và kiểm tra tính hợp lệ của dữ liệu biểu mẫu.

Lấy ví dụ chúng ta có form đăng ký như sau:

ĐĂNG KÝ THÀNH VIÊN	
Họ và tên	
Tên đăng nhập	
Mật khẩu	
Xác nhận mật khẩu	
Giới tính	☐ Nam ☐ Nữ
<input type="submit" value="Đăng ký"/>	

- Luật Validation form có một số nguyên tắc sau:
 - Username: Chỉ bao gồm các ký tự, chữ số, gạch dưới.
 - Email: Bắt buộc là địa chỉ email hợp lệ.
- Phần xử lý:
 - Sử dụng phương thức POST để truyền dữ liệu


```
<form action="<?php echo htmlspecialchars($_SERVER['PHP_SELF']) ?>" method="POST"></form>
```
 - Việc chọn giới tính cho phép người dùng chọn một trong hai lựa chọn


```
<input type="radio" name="gender" value="1"> Nam
          <input type="radio" name="gender" value="0"> Nữ
```
 - Nút submit để người dùng nhấn xác nhận gửi thông tin lên server


```
<input type="submit" value="Đăng ký" name="submit">
```

Với `input type="submit"` khi được người dùng nhấn vào thì đồng nghĩa với việc gửi thông tin trên form lên server.
- Sau đây là một số công việc cần làm
 - Đầu tiên chúng ta cho tất cả các dữ liệu của form qua hàm `htmlspecialchars()`, khi đó các ký tự đặc biệt được chuyển hoàn toàn qua html
 - Loại bỏ các ký tự dư thừa (khoảng trắng) với hàm `trim()`
 - Loại bỏ ký tự gạch chéo (\) với hàm `stripslashes()`
 - Chuyển ký tự đặc biệt về html với hàm `htmlspecialchars()`
- Và công việc cuối cùng viết hàm để check dữ liệu trên form

- Xây dựng hàm *checkInput()*;

```
<?php
function checkInput($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
?>
```

- Xây dựng đoạn code áp dụng để đưa dữ liệu trên form qua bộ lọc bởi hàm *checkInput()*

```
<?php
if($_SERVER['REQUEST_METHOD'] == 'POST'){
    $fullname = $username = $gender = $password = "";
    $fullname = checkInput($_POST['fullname']);
    $username = checkInput($_POST['username']);
    $gender = checkInput($_POST['gender']);
    $password = checkInput($_POST['password']);
}
?>
```

- Việc kiểm tra người dùng submit hay chưa chúng ta sử dụng biến hệ thống *\$_SERVER* để kiểm tra xem phương thức gửi dữ liệu lên SERVER có phải là POST như mình đã khai báo ở FORM hay không?

```
<?php
if($_SERVER['REQUEST_METHOD'] == 'POST'){
    //Code xử lý khi người dùng submit
}
?>
```

B. CÁC DẠNG BÀI TẬP

BT2-1. Xây dựng form login cho phép người sử dụng đăng nhập hệ thống như hình sau đây:

NHẬP THÔNG TIN ĐĂNG NHẬP

Tên đăng nhập:

Mật khẩu:

- + Kiểm tra tính hợp lệ của dữ liệu khi người dùng nhập dữ liệu vào form.

+ Nếu người dùng nhập username='vietkhanh' và password='12345' thì hiển thị trang chu.php. Ngược lại thông báo username và password không tồn tại.

Hướng dẫn giải bài tập 2-1: Sinh viên thiết kế form giao diện và viết chương trình theo code mẫu sau đây:

<?php

```
function checkInput($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}

if($_SERVER['REQUEST_METHOD'] == 'POST'){
    $username = $password = "";
    $username = checkInput($_POST['username']);
    $password = checkInput($_POST['password']);
    if($username=='vietkhanh' && $password=='12345')
        echo "<script language='javascript'>
            location.href='TrangChu.php'; </script>";
    else
        echo "Username hoặc password không hợp lệ";
}
```

?>

```
<form name="frmDangNhap" action="<?php echo
htmlspecialchars($_SERVER['PHP_SELF']) ?>" method="POST">
    <fieldset>
        <legend><h2>NHẬP THÔNG TIN ĐĂNG NHẬP</h2></legend>
        <p><b>Tên đăng nhập:</b><input type="text" name="username"
            size="15" maxlength="25" value="<?php if(isset($_POST['username']))echo
            $username;?>"></p>
        <p><b>Mật khẩu:</b><input type="password" name="password"
            size="22" maxlength="20" value="<?php if(isset($_POST['password']))echo
            $password;?>"></p>
        <div align="center"><input type="submit" name="submit" value="Đăng
            nhập" style="height:inherit"></div>
    </fieldset>
</form>
```

➤ **Một số kết quả khi chạy chương trình:**

Số lần test	username	password	Kết quả
test 1	vietkhanh	12345	Hiển thị trangchu.php
test 2	\vietkhanh	\12345	Hiển thị trangchu.php
test 3	Viet khanh	12345	Hiển thị thông báo username hoặc password không đúng
Test 4	vietkhanh	12345	Hiển thị trangchu.php

C.BÀI TẬP NÂNG CAO

Xây dựng ứng dụng Web cho các bài tập sau đây:

BT2-2. Xây dựng form đăng ký phòng dạy cho phép các giáo sư, giảng viên đăng ký phòng giảng dạy như hình sau đây:

The screenshot shows a web browser window with the URL `localhost/joomla_2.5.5-Stable-Full_Package/`. The page has a blue header with the name 'Tran Viet Khanh' and a navigation bar with links: 'Trang chủ', 'Diễn đàn', 'Tin tức', 'Hỏi đáp', and 'Liên hệ'. On the left, there is a sidebar with a 'Main Menu' and a 'Login Form' containing fields for 'User Name', 'Password', and a 'Remember Me' checkbox. The main content area is titled 'Trang Chủ' and contains a form titled 'ĐĂNG KÝ PHÒNG DẠY'. The form has the following fields: 'Phòng số:' with a dropdown menu showing 'A001', 'Giáo sư giảng dạy:' with a text input field, and 'Ngày / tháng / năm:' with a date picker showing '30 / 11 / 15'. There is a 'Đăng ký' button at the bottom of the form. At the very bottom of the page, it says 'Tran Viet Khanh, Designed By © Tran Viet Khanh - 2015'.

Gợi ý làm bài: Thiết kế form giao diện bao gồm:

- thẻ `<select>`/`</select>` cho phép chọn số phòng.
- thẻ text field cho phép nhập tên giáo sư.
- thẻ `<select>`/`</select>` cho phép chọn ngày, tháng, năm đăng ký.
- Viết code cho nút đăng ký để lưu thông tin đăng ký vào file dạng *.txt

BT2-3. Xây dựng form quản lý nhân viên cho phép tính lương nhân viên như hình sau đây:

The screenshot shows a web browser window with the same URL as the previous image. The page layout is identical, but the main content area now displays a form titled 'QUẢN LÝ NHÂN VIÊN' (Employee Management). The form has a light orange background and contains the following fields: 'Họ và tên:' (Text input), 'Ngày sinh:' (Date input), 'Giới tính:' (Radio buttons for 'Nam' and 'Nữ'), 'Loại nhân viên:' (Radio buttons for 'Văn phòng' and 'Sản xuất'), 'Số ngày vắng:' (Text input), 'Thức lịch:' (Text input), 'Số con:' (Text input), 'Ngày vào làm:' (Date input), 'Hệ số lương:' (Text input), 'Số sản phẩm:' (Text input), 'Tăng ca:' (Radio buttons for 'Có' and 'Không'), and 'Trợ cấp:' (Text input). There is a 'Tính lương' button at the bottom right of the form. The footer remains the same: 'Tran Viet Khanh, Designed By © Tran Viet Khanh - 2015'.

Gợi ý làm bài: Thiết kế form giao diện bao gồm:

- Các thẻ text field cho phép nhập họ tên, ngày sinh, số con, ngày vào làm,...
- thẻ radio cho phép chọn lựa giới tính, loại nhân viên, tăng ca.
- Viết code cho nút tính lương để lưu thông tin lương của nhân viên vào file dạng *.txt

Chương 3

TRUY XUẤT CSDL MYSQL VỚI PHP

A.TÓM TẮT LÝ THUYẾT

§3 : KẾT NỐI VÀ TRUY VẤN CƠ SỞ DỮ LIỆU MYSQL VỚI PHP

3.1. Các bước truy cập cơ sở dữ liệu MySQL

- MySQL là hệ quản trị cơ sở dữ liệu mã nguồn mở.
- PhpMyAdmin là ứng dụng để quản lý hệ quản trị cơ sở dữ liệu MySQL.
- Sử dụng PHP để làm việc với CSDL MySQL thông qua đối tượng PDO (PHP Data Objects)
 - Tạo DSN (Data Source Name) xác định tên máy chủ và tên cơ sở dữ liệu
 - Tạo đối tượng PDO với ba đối số: DSN, tên tài khoản quản trị cơ sở dữ liệu và mật khẩu.
- Sau đây là các bước kết nối và truy xuất dữ liệu:
 - Bước 1: Tạo kết nối đến database server.
`$cn = mysql_connect("tên máy chủ","tên tài khoản","mật khẩu") or die("thông báo");`
Hàm `mysql_connect` với các tham số truyền vào tên máy chủ là "localhost", tên tài khoản là "root", và mật khẩu.
Hàm `die` ("thông báo"): đưa ra thông báo kết thúc.
Với cách viết trên `die` chỉ thực hiện khi lệnh trước nó không thành công.
 - Bước 2: Lựa chọn cơ sở dữ liệu.
`mysql_select_db("tên csdl") or die("chưa có csdl");`
 - Bước 3: Xây dựng truy vấn và thực hiện truy vấn
`$bien = mysql_query("lệnh sql") or die("không thực hiện được lệnh sql");`
 - Bước 4: Xử lý kết quả trả về.
 - `mysql_affected_rows()`: dùng kiểm tra data có thay đổi trên bao nhiêu dòng, dùng trong trường hợp INSERT, UPDATE, REPLACE, hoặc DELETE đã làm thay đổi số dòng là bao nhiêu.
 - `$num_rows = mysql_num_rows()`: trả về số lượng các dòng lấy được từ bảng.
 - `$r = mysql_fetch_array($rs)`: trả về 1 dòng của \$rs vào \$r. Tiến hành lấy thông tin từng dòng của bảng .

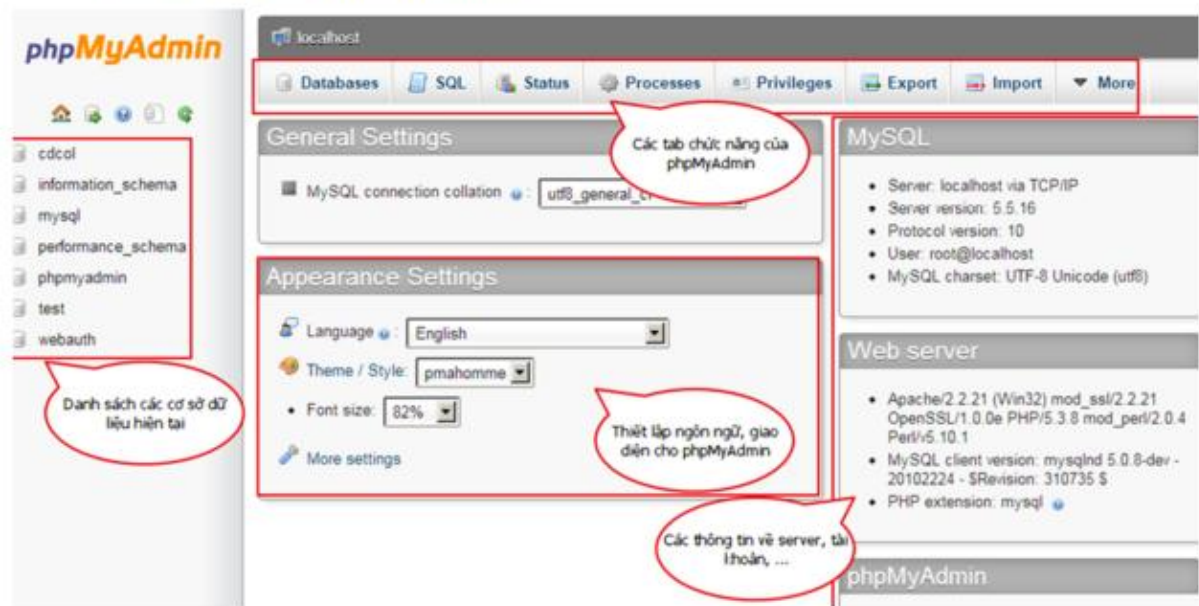
mysql_error(): thông báo lỗi (nếu có).

- Bước 5: Đóng kết nối đến server.

3.2. Sử dụng cơ sở dữ liệu MySQL

- Khởi động AppServer/WampServer/XAMPP
- Chọn phpMyAdmin

■ Giao diện của phpMyAdmin:



- Xem dữ liệu và cấu trúc bảng



3.3. Thực hiện các câu truy vấn

Các câu lệnh SQL được bắt đầu với **INSERT**, **UPDATE**, **DELETE** đều không trả lại dữ liệu mà chỉ cho biết thực hiện thành công hay không. Ngược lại câu truy vấn **SELECT** thường trả lại dữ liệu cần được xử lý bằng hàm **mysql_fetch_array()**. Hàm này nhận vào biến kết quả của câu truy vấn (biến **\$result**) và trả lại từng mẫu tin ở dạng mảng. Chúng ta thường sử dụng hàm này trong một vòng lặp để truy xuất từng mẫu tin được trả lại. Kết cấu cơ bản của việc đọc ra từng mẫu tin của một câu truy vấn là:

```
while ($row = mysql_fetch_array($result)){  
    // thực hiện công việc gì đó với biến $row  
}
```

Hàm **mysql_fetch_array()** nhận vào một tham số tùy biến cho biết loại mảng sẽ được trả lại: liên kết, chỉ mục hoặc cả hai. Mảng liên kết cho phép truy xuất đến các giá trị cột theo tên (**\$row["tên cột"]**), trong khi mảng chỉ mục sử dụng các chỉ số (bắt đầu với 0 là chỉ số của cột đầu tiên **\$row[0]**).

Các hằng được dùng với hàm **mysql_fetch_array()** để quy định cách trả lại mảng kết quả. Giá trị mặc định là **MYSQL_BOTH**

Hằng	Ví dụ
MYSQL_ASSOC	\$row["tên cột"]
MYSQL_NUM	\$row[0]
MYSQL_BOTH	\$row[0] hoặc \$row["tên cột"]

Nhiều người dùng **MYSQL_NUM** vì ít chiếm bộ nhớ nhất. Một số developer khác sử dụng **MYSQL_ASSOC** cách truy xuất này vẫn làm việc tốt khi cấu trúc dữ liệu bị thay đổi. Giá trị mặc định khi sử dụng hàm **mysql_fetch_array()** là **MYSQL_BOTH**.

Lưu ý: Nên giải phóng tài nguyên(kết quả) của hàm **mysql_fetch_array()** khi không còn sử dụng đến nó nữa. Hàm **mysql_free_result(\$result)**;

Một số hàm tương đương với hàm **mysql_fetch_array()**:

- Hàm **mysql_fetch_row()** tương đương với **mysql_fetch_array(\$result, MYSQL_NUM)**.
- Hàm **mysql_fetch_assoc()** tương đương với **mysql_fetch_array(\$result, MYSQL_ASSOC)**.
- PHP cũng có hàm **mysql_fetch_object()** trả lại một mẫu tin dưới dạng một đối tượng.

3.4. An toàn cơ sở dữ liệu

An toàn cơ sở dữ liệu trong PHP được chia thành 2 nội dung chính:

- **Bảo vệ thông tin khi truy xuất cơ sở dữ liệu:**
 - Không cho người dùng không rõ danh tính truy cập cơ sở dữ liệu.
 - Luôn bắt buộc phải có mật khẩu khi kết nối đến cơ sở dữ liệu MySQL.
 - Yêu cầu người dùng xác định địa chỉ máy. Điều này giới hạn người dùng có thể truy xuất cơ sở dữ liệu MySQL
 - Gán cho mỗi người dùng một quyền truy cập nhất định ở mức tối thiểu.
 - Chỉ cho phép tài khoản root được truy xuất từ máy cục bộ (localhost)
- **An toàn khi chèn dữ liệu vào cơ sở dữ liệu:**
 - Khi chứa các thông tin nhạy cảm, đặc biệt là mật khẩu, hãy mã hóa dữ liệu trước khi chèn nó vào cơ sở dữ liệu bằng cách sử dụng hàm PASSWORD() hoặc ENCRYPT () hoặc MD5().
 - Không giống với hàm PASSWORD, hàm ENCRYPT hoặc MD5 nó cũng mã hóa một chuỗi nhưng quá trình mã hóa có tính chất ngẫu nhiên, không thể giải mã được.

Ví dụ:

```
+ insert into users(username,password) values('tên user',PASSWORD('mật khẩu'));
+ insert into users(username,password) values('tên user',ENCRYPT('mật khẩu',
'salt')); tham số salt để làm cho quá trình mã hóa có tính ngẫu nhiên.
+ insert into users(username,password) values('tên user',md5('mật khẩu'));
```

§4 : TẠO FORM HTML - CSS NHẬP LIỆU VÀ DÙNG PHP GIAO TIẾP VỚI CSDL

4.1. Cấu trúc site

Khi bắt đầu sử dụng nhiều tập tin trong các ứng dụng Web, cấu trúc tổng thể của một site rất quan trọng. Khi thực hiện bố cục site, cần quan tâm đến các vấn đề sau:

- Dễ bảo trì.
- Dễ truy cập
- An toàn dữ liệu
- Thiết kế website sao cho người dùng dễ định hướng theo cả hai cách là nhấp chuột hoặc nhập trực tiếp URL. Tránh tạo ra quá nhiều thư mục lồng vào nhau.

4.2. Tạo và gọi các hàm

Cú pháp để tạo ra một hàm là:

```
function function_name($param_1, $param_2, ..., $param_n){
    // mã xử lý
    [ return return_value]
}
```

Để viết hàm có trả về dữ liệu, viết câu lệnh return cuối thân hàm. Lệnh return kết thúc việc thực hiện hàm và trả về giá trị xác định. Nếu không gán giá trị trả về thì giá trị NULL sẽ được trả về

Để hàm không trả về dữ liệu, không viết lệnh return.

Khi thực hiện lời gọi hàm, các đối số trong danh sách đối số phải theo cùng thứ tự với các tham số trong danh sách tham số mà hàm đã xác định và phải tương thích về kiểu dữ liệu.

Hướng dẫn xây dựng hàm:

// Hàm không có tham số và trả về một giá trị

```
function coin_toss(){
    $coin = (mt_rand(0,1)==0)? 'heads': 'tail';
    return $coin;
}
```

// Hàm có tham số và trả về một giá trị

```
function avg_of_3($x, $y, $z){
    $avg = ($x+$y+$z)/3;
    return $avg;
}
```

// Hàm có tham số và không trả về giá trị

```
function display_error($error){
    echo '<p class= "error">'.$error.'</p>';
}
```

Hướng dẫn gọi hàm:

// Hàm trả về giá trị

```
$average = avg_of_3(5, 2, 8);
```

```
echo coin_toss();
```

// Hàm không trả về giá trị

```
display_error('Value out of range.');
```

4.3. Thiết lập các tham số mặc định

Cú pháp:

<Tên tham số> = <giá trị mặc định>

Giá trị mặc định phải là giá trị hoặc mảng các giá trị vô hướng hoặc là giá trị NULL.

// Hàm với một tham số mặc định 'coin'

```
function get_rand_bool_text($type = 'coin'){
    $rand = mt_rand(0,1);
    switch ($type) {
        case 'coin':
            $result = ($rand == 1)? 'heads': 'tails';
            Break;
        case 'switch':
            $result = ($rand == 1)? 'on': 'off';
            Break;
    }
    return $result;
}
```

// Hàm với một tham số bắt buộc và hai tham số mặc định 'p', 'error'

```
function display_error($error, $tag = 'p', $class= 'error'){
    $opentag = '<'.$tag.' class= "'.$class.'">';
    $closetag = '</'.$tag.'>';
    echo $opentag.$error.$closetag;
}
```

§5 : MÔ HÌNH MASTER – SINGLE TRONG VIỆC XÂY DỰNG WEBSITE

5.1. Giới thiệu mô hình Master – Single.

5.2. Sử dụng Web Editor trong việc biên tập nội dung web phức tạp.

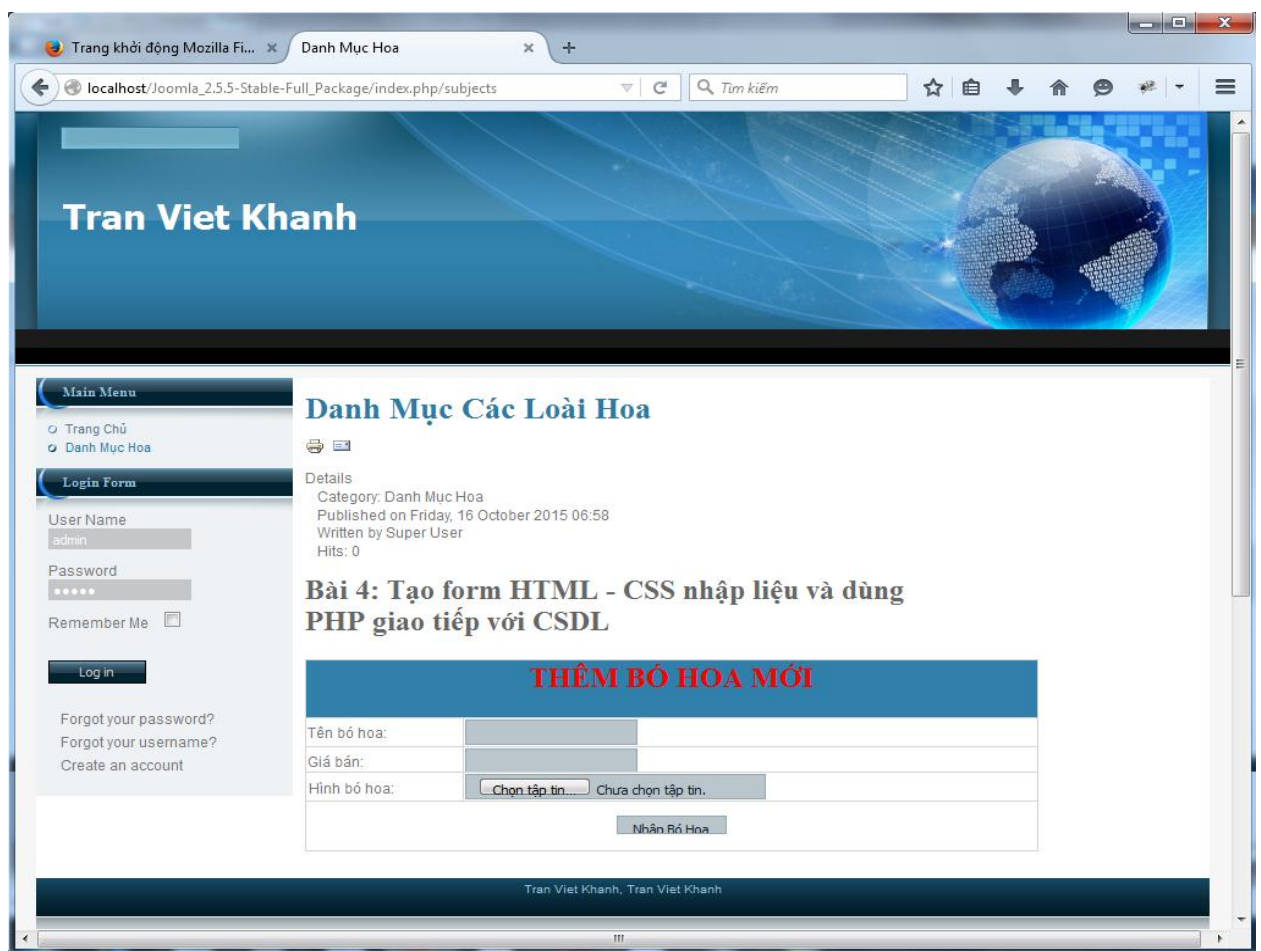
5.3. Phân loại dữ liệu, liệt kê dữ liệu phân loại trong PHP.

5.4. Phân trang dữ liệu.

5.5. Lọc và tìm kiếm dữ liệu.

B.CÁC DẠNG BÀI TẬP

BT3-1. Xây dựng form cho phép người sử dụng thêm thông tin bó hoa vào CSDL như hình sau đây:



Hướng dẫn giải bài tập 3-1: Sinh viên thiết kế form giao diện và viết chương trình theo code mẫu sau đây:

```
<?php
if(isset($_POST["ten_hoa"])&&isset($_POST["gia_ban"]))
{
    if(($_FILES["file"]["type"]=="image/gif")||
        ($_FILES["file"]["type"]=="image/jpeg")&& ($_FILES["file"]["size"]<20000))
    {
```

```

if($_FILES["file"]["error"]>0)
{
    echo "Return Code:".$_FILES["file"]["error"]."<br>";
}
else
{
    move_uploaded_file($_FILES["file"]["tmp_name"],
    "templates/siteground-j16-10/hinh_anh/ ".$_FILES["file"]["name"]);
    $duong_dan="templates/siteground-j16-10/hinh_anh/
    ".$_FILES["file"]["name"];
}
$ten_hoa=$_POST["ten_hoa"];
$gia_ban=$_POST["gia_ban"];
$hinh_anh=$_FILES["file"]["name"];
$cn = mysql_connect("localhost","root","") or die("Couldn't to open
database");
mysql_select_db("quanlybanhoa");
$sql = "insert into hoa values ('".$_ten_hoa."', '".$_gia_ban."', '".$_hinh_anh.'")";
mysql_query($sql,$cn);

echo "<p align='center'><b>Successful !</b></p>";
mysql_close($cn);
}
else
    echo "Invalid file";
}
else
{
    echo "<p align='center'>Input data, please!</p>";
}
?>
<form name="frmNhapBoHoa" method="post" action="<?php echo
htmlspecialchars($_SERVER['PHP_SELF']) ?>" enctype="multipart/form-data">
    <table width="600" border="1">
        <tr>
            <td colspan="2" align="center" valign="middle"
            bgcolor="#3180AB"><h1 class="style1">TH&Ecirc;M B&Oacute;
            HOA M&#7898;I </h1></td>
        </tr>
        <tr>
            <td>T&ecirc;n b&oacute; hoa:</td>
            <td><input type="text" name="ten_hoa" /></td>
        </tr>
        <tr>
            <td>Gi&aacute; b&aacute;n:</td>
            <td><input type="text" name="gia_ban" /></td>
        </tr>
        <tr>
            <td>H&igrave;nh b&oacute; hoa:</td>
            <td><input type="file" name="file"/></td>

```

```

</tr>
<tr>
    <td colspan="2" align="center" valign="middle" height="40">
        <input type="submit" name="btnNhap" value="Nhập Bó Hoa"
        height="30"/></td>
</tr>
</table>
</form>

```

➤ Một số kết quả của bảng hoa trong csdl khi chạy chương trình:

The screenshot shows the phpMyAdmin interface. On the left, the database 'quanlybanhoa' is selected, and the table 'hoa' is highlighted. The main area displays the table structure and data. The table has 3 rows and 4 columns: 'mahoa', 'tenhoa', 'giaban', and 'hinhanh'. The data is as follows:

	mahoa	tenhoa	giaban	hinhanh
☐	1	hoa cuoi	12334	hinh1.gif
☐	31	Hoa Mong Ga	90000	hoamongga.jpg
☐	30	Lan dendro	25000	landendro.jpg

C.BÀI TẬP NÂNG CAO