

**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**

**HƯỚNG DẪN SỬ DỤNG NGÔN NGỮ LẬP TRÌNH
PHP TỰ HỌC XÂY DỰNG CÁC ỨNG DỤNG WEB**

BIÊN SOẠN ĐỀ TÀI: Trần Việt Khánh

ĐẶT VẤN ĐỀ

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.

*Do là khối kiến thức chuyên ngành nên khi bắt tay vào thực hiện xây dựng và phát triển một ứng dụng Web bằng ngôn ngữ lập trình PHP sinh viên thường gặp khó khăn trong việc chuyển tiếp các vấn đề trên lý thuyết sang lập trình cụ thể trên máy tính. Có rất nhiều tài liệu hướng dẫn lập trình Web bằng ngôn ngữ PHP trong nước và trên thế giới cũng như trên mạng internet, nhưng hầu như các tài liệu này chỉ tập trung trình bày về mặt lý thuyết và ít hướng dẫn về phương pháp thực hành để gần với thực tiễn thị trường làm trang Web. Vì vậy sinh viên rất khó khăn trong việc tự học cũng như tiếp cận các vấn đề thực tế trong thị trường làm Web, đề tài **“Hướng dẫn sử dụng ngôn ngữ lập trình PHP tự học xây dựng các ứng dụng Web”** nhằm cung cấp cho sinh viên những kỹ năng thực hành cơ bản và nâng cao về ngôn ngữ lập trình PHP, đồng thời giúp cho sinh viên nâng cao khả năng tự học, hoàn thiện dần các kỹ năng lập trình, phong cách viết chương trình, tiến tới xây dựng các ứng dụng Web một cách hoàn chỉnh. Đề tài cũng giúp cho giảng viên thuận tiện trong việc giảng dạy lý thuyết và giảng dạy thực hành, thuận tiện trong việc đánh giá kết quả học tập của sinh viên. Giúp cho giảng viên hoàn thiện hơn về kỹ năng lập trình ngôn ngữ PHP, khả năng trình bày, đồng thời giúp cho việc trao đổi chuyên môn của các giảng viên dễ dàng hơn. Từ đó có thể được nhân rộng ra để biên soạn các đề tài khác nhằm mục đích nâng cao chất lượng đào tạo sinh viên của trường.*

Đề tài này được trình bày bao gồm 4 chương liên quan đến ngôn ngữ lập trình PHP và cơ sở dữ liệu MySQL. Mỗi một chương 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 bài của từng chương 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 bài tập nâng cao có hướng dẫn cho sinh viên 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.

Đề tài này được biên soạn để hỗ trợ cho việc giảng dạy học phần lập trình PHP ở bậc cao đẳng. Hệ thống bài tập này cũng có thể làm tài liệu tham khảo cho các giảng viên đang giảng dạy kiến thức lập trình PHP ở các bậc đào tạo khác. Tùy theo nội dung chương trình đào tạo mà giáo viên có thể chọn một số nội dung phù hợp trong đề tài này để giảng dạy.

*Tôi trân trọng giới thiệu với bạn đọc **“Hướng dẫn sử dụng ngôn ngữ lập trình PHP tự học xây dựng các ứng dụng Web”** này và hy vọng rằng nó sẽ giúp cho việc giảng dạy và tự học môn lập trình PHP được thuận lợi hơn.*

Cuối cùng, 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**

Các Khái Niệm PHP Cơ Bản

- 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 SĐT và gán giá trị cho nó là 0909090909

```
define('SĐT', '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	<code>\$a + \$b</code>
Phép trừ	-	Trừ hai số hạng	<code>\$a - \$b</code>
Phép nhân	*	Nhân hai số hạng	<code>\$a * b</code>
Phép chia	/	Chia hai số hạng	<code>\$a / \$b</code>
Phép chia lấy dư	%	Chi lấy dư	<code>5 % 2 = 1</code>

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	<code>==</code>	Hai số hạng bằng nhau	<code>\$a == \$b</code>
So sánh khác	<code>!=</code>	Hai số hạng khác nhau	<code>\$a != \$b</code>
So sánh lớn hơn	<code>></code>	So sánh lớn hơn	<code>\$a > b</code>
So sánh nhỏ hơn	<code><</code>	So sánh nhỏ hơn	<code>\$a < \$b</code>
Lớn hơn hoặc bằng	<code>>=</code>	Lớn hơn hoặc bằng	<code>\$a >= \$b</code>
Nhỏ hơn hoặc bằng	<code><=</code>	Nhỏ hơn hoặc bằng	<code>\$a <= \$b</code>

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 thỏa mãn điều kiện	$\$a > \$b \ \&\& \ \$a > \c
Phép Or		Một trong 2 vế thỏa 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 thức2>)  
    <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 tinhcong($so1, $so2);
```

- **Truyền tham biến và tham trị.**

Việc gọi hàm tính tổng (`tinhcong($so1,$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 "
 Trong hàm F(), a = ".$a." và b = ".$b;
```

```
}
```

```
$a = 8;
```

```
$b = 8;
```

```
echo "
 1. Ngoài hàm F(), a = ".$a." và b = ".$b;
```

```
//Gọi hàm tham trị.
```

```
F($a, $b);
```

```
echo "
2. Ngoài hàm F(), a = ".$a." và b = ".$b;
```

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

```
F(&$a, $b);
```

```
echo "
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.

## Các Khái Niệm PHP Cơ Bản

---

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!
";
 echo "Tên file : ".$name."
";
 echo "Kiểu file : ".$type."
";
 echo "File size : ".$size;
 }
?>
```

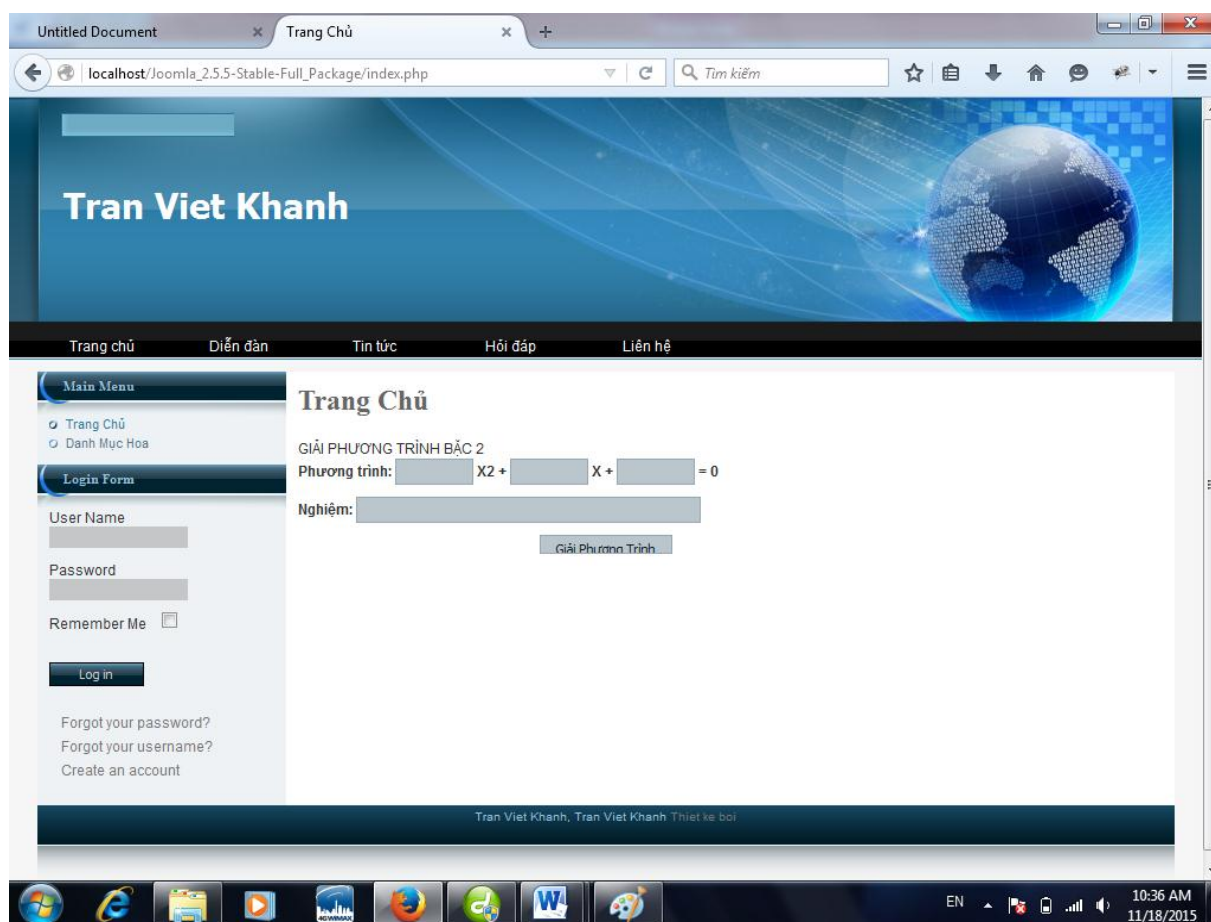
## 1.8. Tạo trang PHP đầu tiên.

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);
}
```

## Các Khái Niệm PHP Cơ Bản

??

```
<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="post">
 <fieldset><legend>GIẢI PHƯƠNG TRÌNH BẬC 2</legend>
 <p>Phương trình:
 <input type="text" name="a" size="5" maxlength="5" value="<?php
 if(isset($_POST['a'])) echo $_POST['a'];?>" />
 X2 +
 <input type="text" name="b" size="5" maxlength="5" value="<?php
 if(isset($_POST['b'])) echo $_POST['b'];?>" />
 X +
 <input type="text" name="c" size="5" maxlength="5" value="<?php
 if(isset($_POST['c'])) echo $_POST['c'];?>" />
 = 0
 </p>
 <p>Nghiem:
 <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 số 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  $\geq 0$  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");
$tong=NULL;
function Sn($n)
{
 for($i=0;$i<=$n;$i++)
 $tong=$tong+$i;
 return $tong;
}
if(isset($_POST["n"]))
{
 $n=$_POST["n"];
```

## Các Khái Niệm PHP Cơ Bản

---

```
$tong=Sn($n);
```

```
}
```

```
?>
```

```
<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="post">
```

```
<fieldset><legend>Tính tổng $S(n) = 1 + 2 + \dots + n$ với n là số tự nhiên ≥ 0 </legend>
```

```
<p>Nhập n:<input type="text" name="n" size="5" maxlength="5" value="<?php if(isset($_POST['n'])) echo $_POST['n'];?>" /></p>
```

```
<p>S(n):
```

```
<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>
```

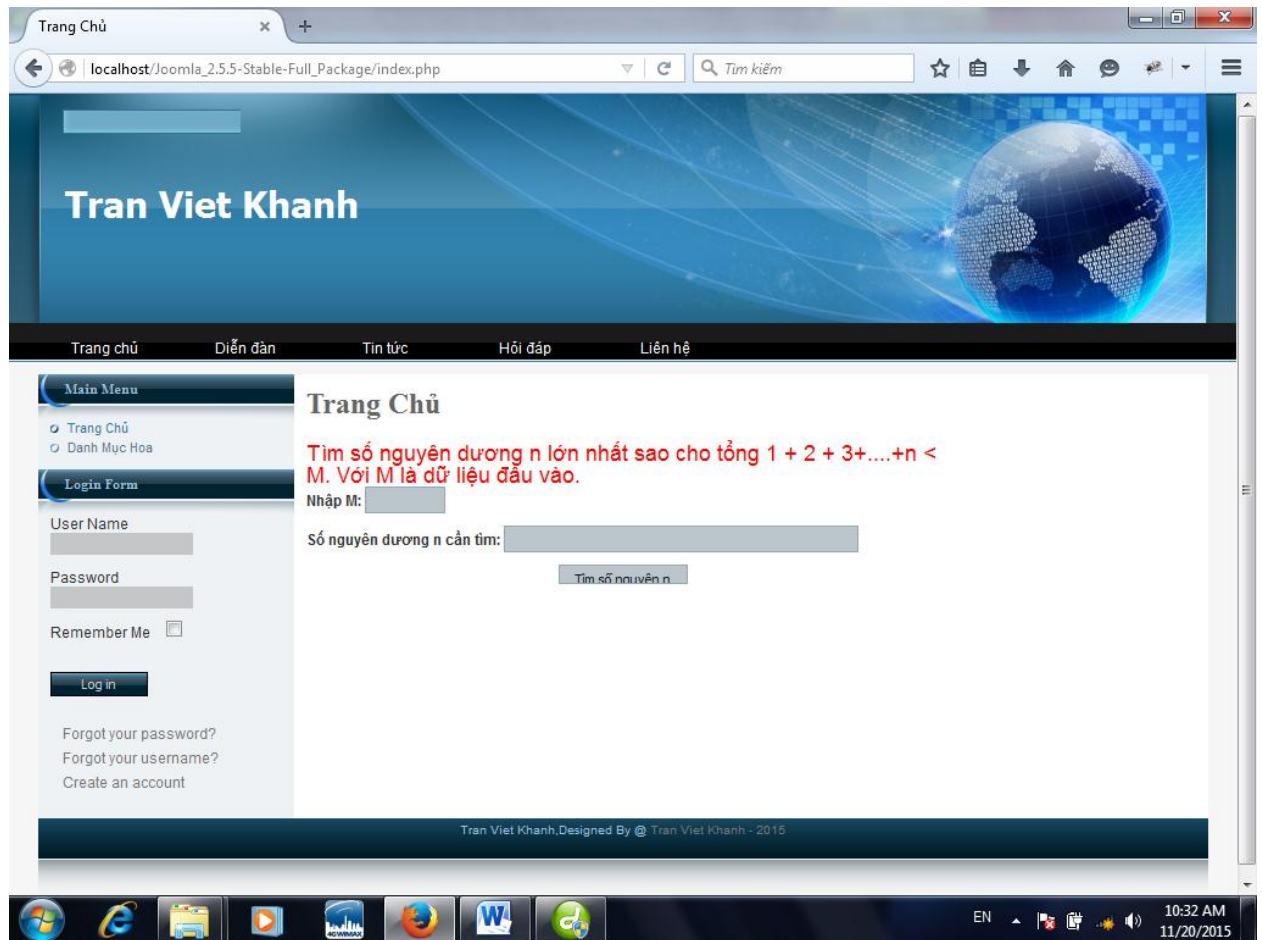
```
</form>
```

➤ **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 $_SERVER['PHP_SELF'];?>" method="post">
<fieldset><legend>Tìm số nguyên dương n lớn nhất
sao cho tổng 1 + 2 + 3 + ... + n < M. Với M là dữ liệu đầu vào.</legend>
<p>Nhập M:
<input type="text" name="M" size="5" maxlength="5" value="<?php
if(isset($_POST['M'])) echo $_POST['M'];?>" />
</p>
<p>Số nguyên dương n cần tìm:
<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:

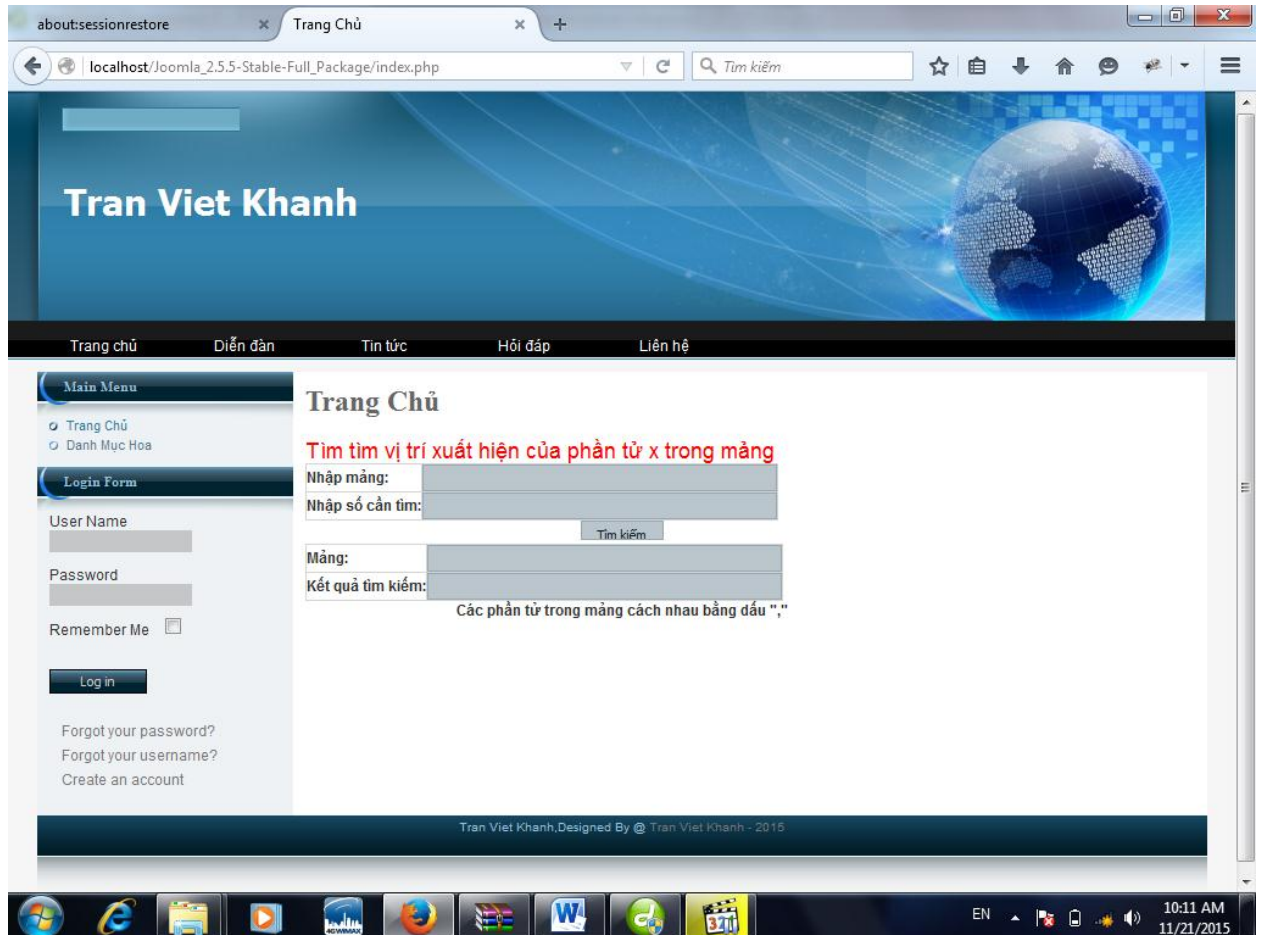
| Số lần test | test 1 | test 2 | test 3 | test 4 |
|-------------|--------|--------|--------|--------|
| M           | 15     | 20     | 100    | 1000   |

## Các Khái Niệm PHP Cơ Bản

|   |   |   |    |    |
|---|---|---|----|----|
| 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 $_SERVER['PHP_SELF'];?>" method="post">
 <fieldset><legend>Tìm tìm vị trí xuất hiện của
 phần tử x trong mảng</legend>
 <table border="1">
 <tr>
 <td>Nhập mảng:</td>
 <td><input type="text" name="mang" size="50" maxlength="50"
 value="<?php if(isset($_POST['mang'])) echo $_POST['mang']; ?>" /></td>
 </tr>
 <tr>
 <td>Nhập số cần tìm:</td>
 <td><input type="text" name="x" size="50" maxlength="50" value="<?php
 if(isset($_POST['x'])) echo $_POST['x'];?>" /></td>
 </tr>
 </table>
 <div align="center"><input type="submit" name="submit" value="Tìm kiếm"/></div>
</table border="1">
```

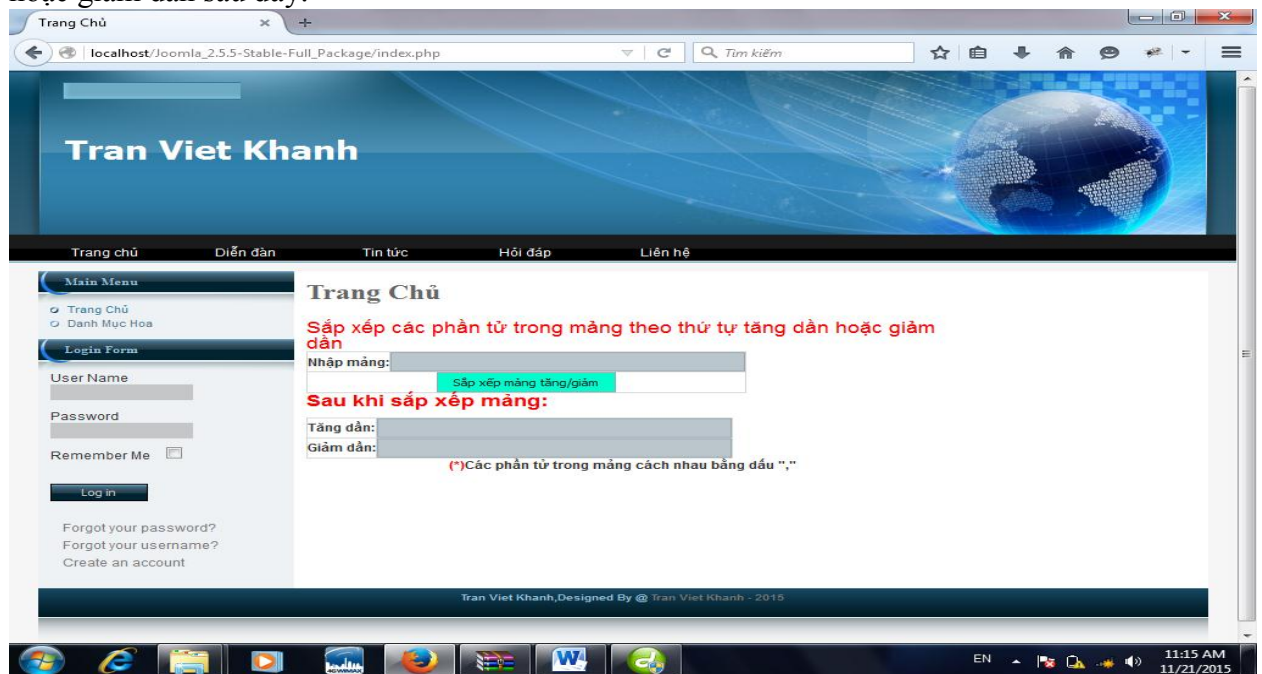
## Các Khái Niệm PHP Cơ Bản

```
<tr>
 <td>Mảng:</td>
 <td><input type="text" name="xuatmang" size="50" maxlength="50"
 value="<?php echo $mangketqua;?>" /></td>
</tr>
<tr>
 <td>Kết quả tìm kiếm:</td>
 <td><input type="text" name="vt" size="50" maxlength="50" value="<?php
 echo $chuoi;?>" /></td>
</tr>
</table>
<div align="center">Các phần tử trong mảng cách nhau bằng dấu "," </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:



## Các Khái Niệm PHP Cơ Bản

---

**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>Sắp xếp các phần tử trong mảng theo
 thứ tự tăng dần hoặc giảm dần</legend>
 <table border="1">
```

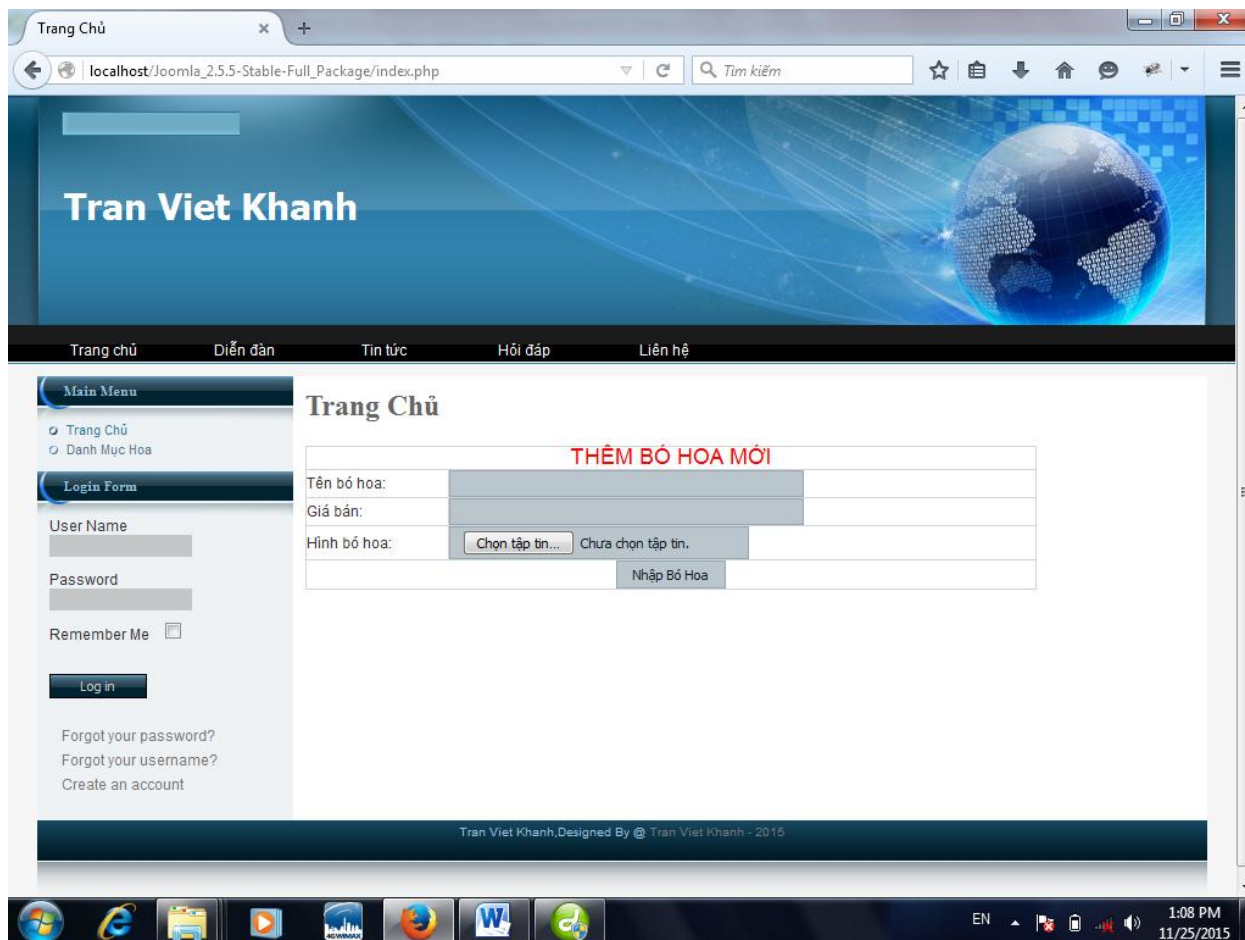
```
<tr>
 <td>Nhập mảng:</td>
 <td><input type="text" name="mang" size="50" maxlength="50" value="<?php
 if(isset($_POST['mang'])) echo $_POST['mang'];?>" /></td>
</tr>
<tr>
 <td colspan="2"><input type="submit"
 style="background-color:#00FFCC; height:inherit" name="submit" value="Sắp xếp
 mảng tăng/giảm"/></td>
</tr>
</table>
<p>Sau khi sắp xếp mảng:</p>
<table border="1">
<tr>
 <td>Tăng dần:</td>
 <td><input type="text" name="mangtang" size="50" maxlength="50"
 value="<?php echo $chuoitang;?>" /></td>
</tr>
<tr>
 <td>Giảm dần:</td>
 <td><input type="text" name="manggiam" size="50" maxlength="50"
 value="<?php echo $chuoigiảm;?>" />
 </td>
</tr>
</table>
<div align="center">(*)Các phần tử trong mảng
 ánh nhau bằng dấu "," </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"]."
";
 }
 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,$noidung);
 echo "<p align='center'>Đã ghi file thành công !</p>";
 fclose($file);
 // B2: Xuất bó hoa vừa ghi ra màn hình
 $gia_ban = number_format($gia_ban,0,',',');
 echo "<p align='center'>Kết quả:

Tên bó
hoa:$ten_hoa
Giá bán:$gia_ban 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">THÊM BÓ HOA
MỖI</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:

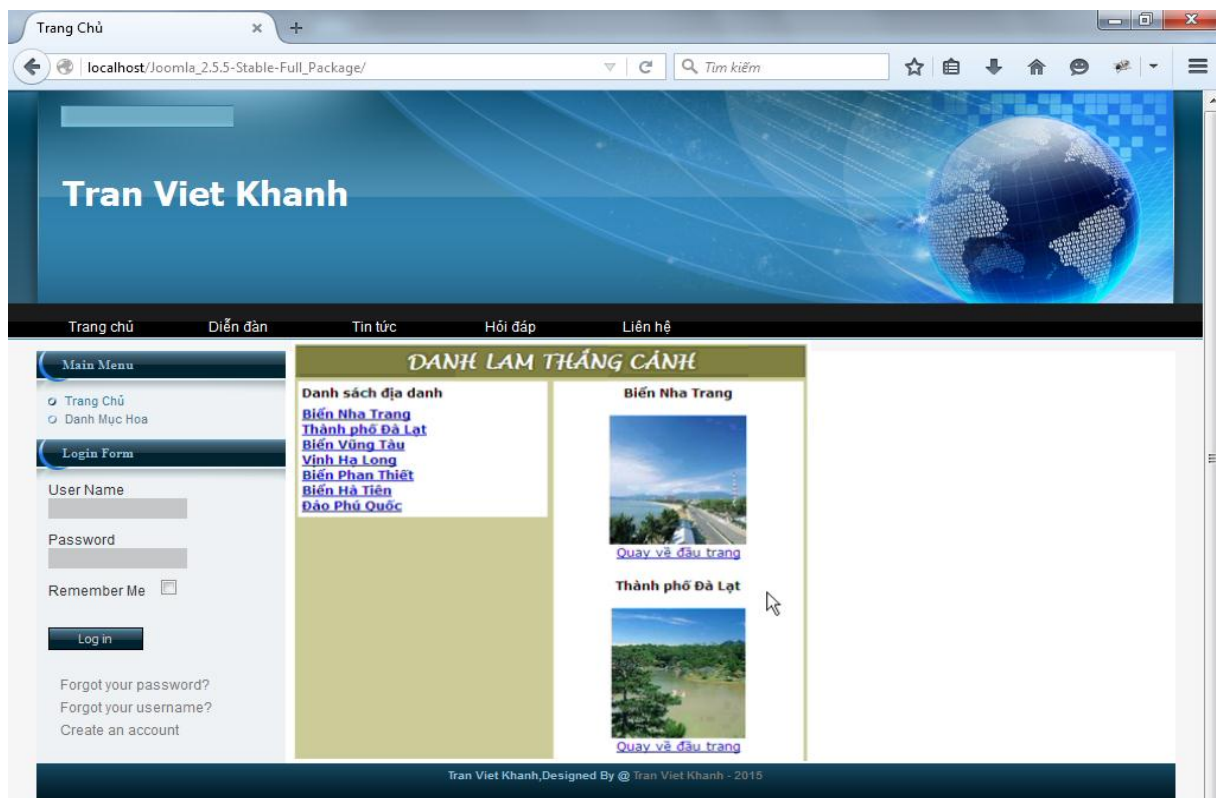
- Textfield cho phép nhập số tự nhiên  $n$ .
- Textfield hiển thị kết quả  $n$  có là số nguyên tố hay không?
- Lấy giá trị biến  $n$  trên form thông qua biến  $$_POST$ .
- Viết hàm kiểm tra  $n$  có là số nguyên tố hay không (kết hợp if và vòng lặp for).
- Nếu  $n$  là số nguyên tố hàm cho kết quả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

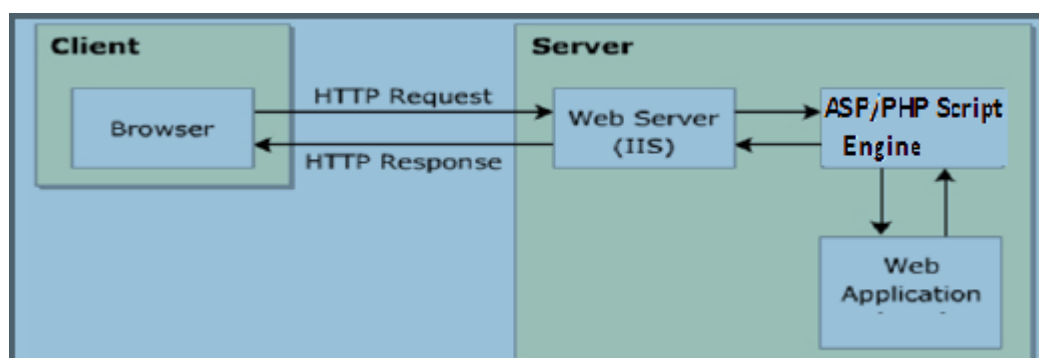
#### §1: CÁC PHƯƠNG PHÁP GIAO TIẾP GIỮA NGƯỜI DÙNG VÀ MÁY CHỦ

##### 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.

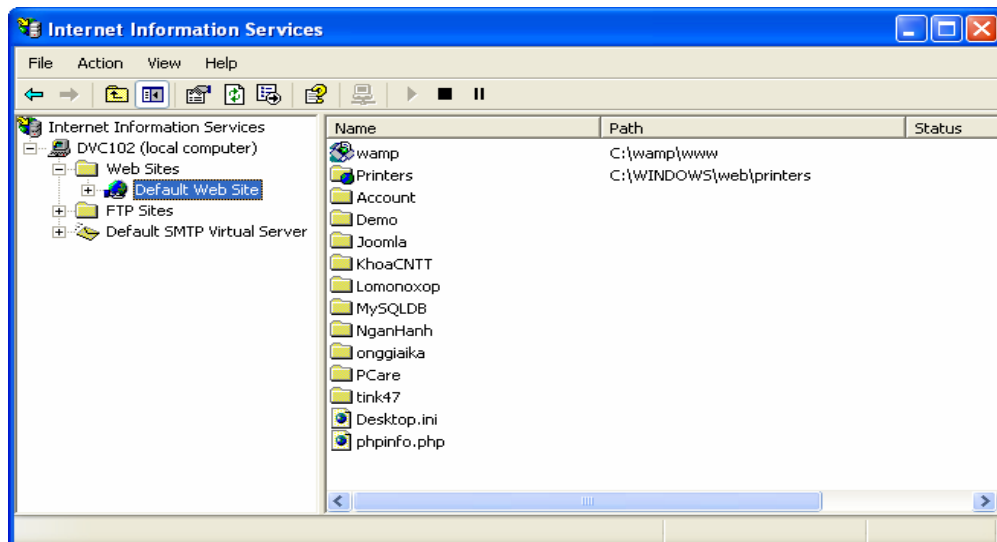
Web động và Web tĩnh: Ứng dụng Web luôn tồn tại hai loại là trang Web động và tĩnh. Trang Web tĩnh là trang HTML không có kết nối cơ sở dữ liệu. Ngược lại, trang Web động là trang Web có kết nối cơ sở dữ liệu. Điều này có nghĩa là mỗi khi trang Web động được làm mới dữ liệu trình bày trên trang Web nó 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,...

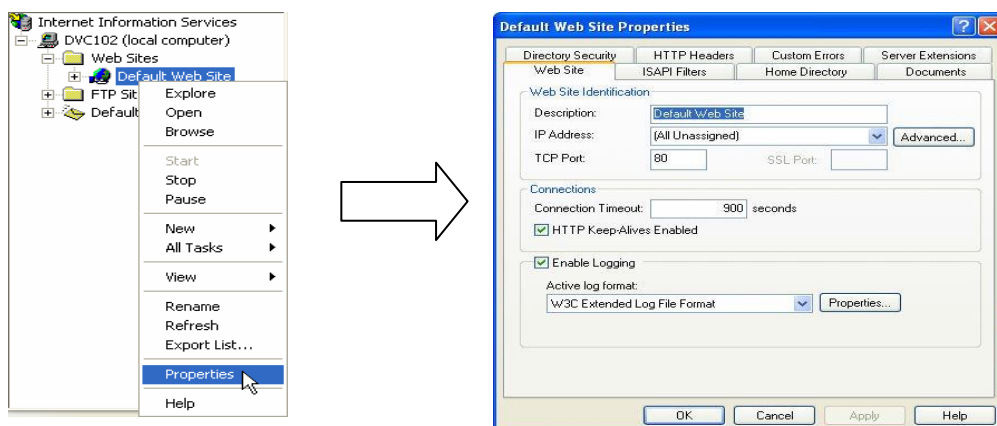
Trong môi trường Linux, Web Server thường sử dụng như Apache, JRun, Web logic,...

### 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.

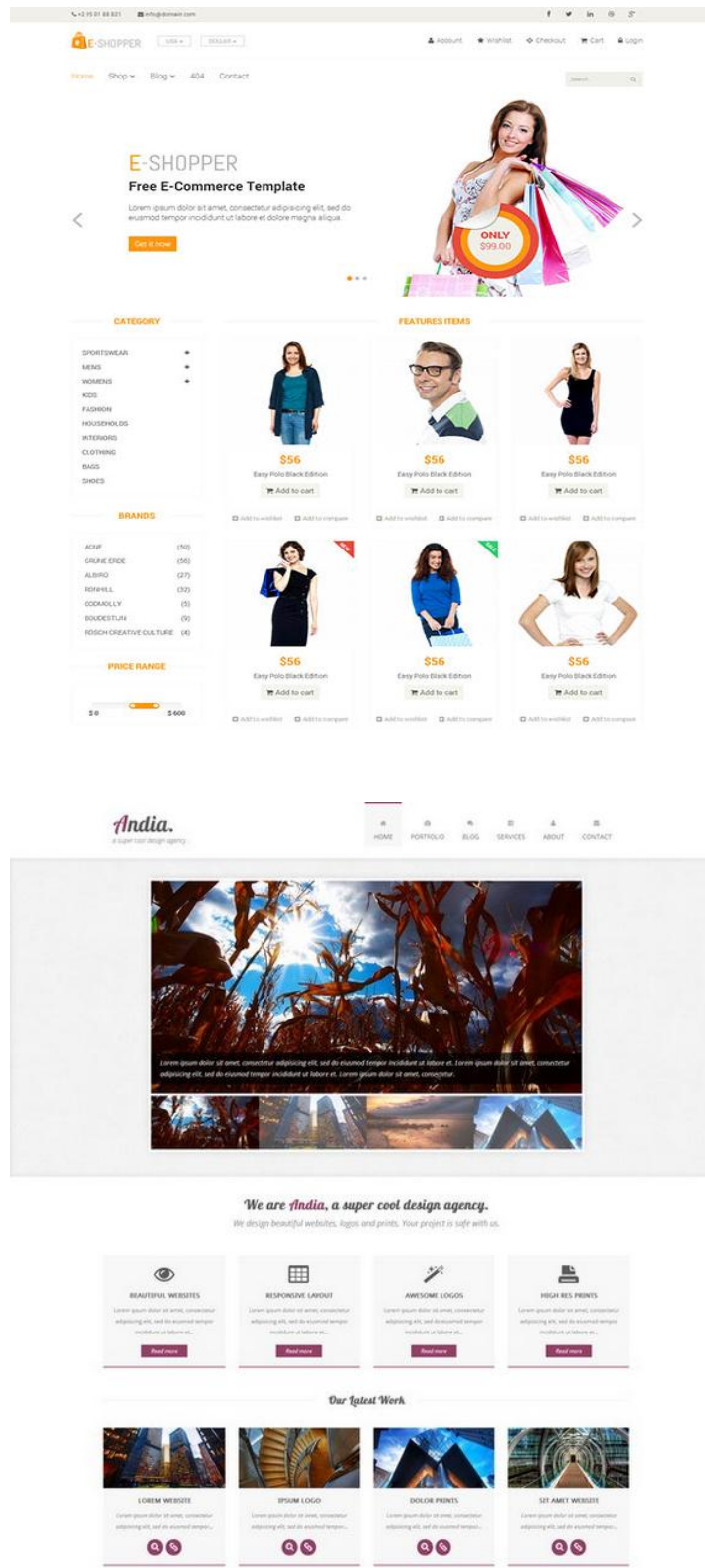
*Khác với HTML, HTML5 có thêm một số các tag markup mới sau đây:*

- **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:*



# Thiết Kế Giao Diện HTML - CSS



### 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	<input type="radio"/> Nam <input type="radio"/> 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 (\) 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:

+ 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
if($_SERVER['REQUEST_METHOD'] == 'POST'){
 $username = $password = "";
 $username = $_POST['username'];
 $password = $_POST['password'];
 if($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>Tên đăng nhập:<input type="text" name="username"
size="15" maxlength="25" value="<?php if(isset($_POST['username']))echo
$username;?>"></p>
 <p>Mật khẩu:<input type="password" name="password"
size="22" maxlength="20" value="<?php if(isset($_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	<input type="checkbox"/> vietkhanh	12345	Hiện thị trangchu.php
test 2	\vietkhanh	\12345	Hiện thị trangchu.php
test 3	Viet khanh	12345	Hiện thi thông báo username hoặc password không đúng
Test 4	vietkhanh	12345	Hiện 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 Joomla! website interface. The main header features the name 'Tran Viet Khanh' and a navigation menu with links: Trang chủ, Diễn đàn, Tin tức, Hỏi đáp, Liên hệ. Below the header, there is a 'Main Menu' section with links to Trang Chủ and Danh Mục Học. The 'Login Form' section includes fields for User Name, Password, and a 'Remember Me' checkbox, with 'Log in' and 'Forgot your password?' links. The 'Đăng ký PHÒNG DẠY' (Register Teaching Room) form is the central focus, containing a dropdown for 'Phòng số: A001', a text field for 'Giáo sư giảng dạy:', a date picker for 'Ngày / tháng / năm: 30 / 11 / 15', and a 'Đăng ký' button. At the bottom, there are links for 'Forgot your password?', 'Forgot your username?', and 'Create an account'. The footer text reads '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 URL `localhost/joomla_2.5.5-Stable-Full_Package/`. The page has a 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' containing 'Trang Chủ' and 'Danh Mục Hoa', and a 'Login Form' with fields for 'User Name' and 'Password', a 'Remember Me' checkbox, and a 'Log in' button. Below the login form are links for 'Forgot your password?', 'Forgot your username?', and 'Create an account'. The main content area is titled 'Trang Chủ' and contains a form titled 'QUẢN LÝ NHÂN VIÊN'. The form has the following fields: 'Họ và tên:', 'Số con:', 'Ngày sinh:', 'Ngày vào làm:', 'Giới tính:' (with radio buttons for 'Nam' and 'Nữ'), 'Hệ số lương:', 'Loại nhân viên:' (with radio buttons for 'Văn phòng' and 'Sân xuất'), 'Số ngày vắng:', 'Số sản phẩm:', 'Tăng ca:' (with radio buttons for 'Có' and 'Không'), 'Tiền lương:', 'Trợ cấp:', and 'Thực lĩnh:'. There is a 'Tính lương' button at the bottom of the form. The footer of the page says '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 .

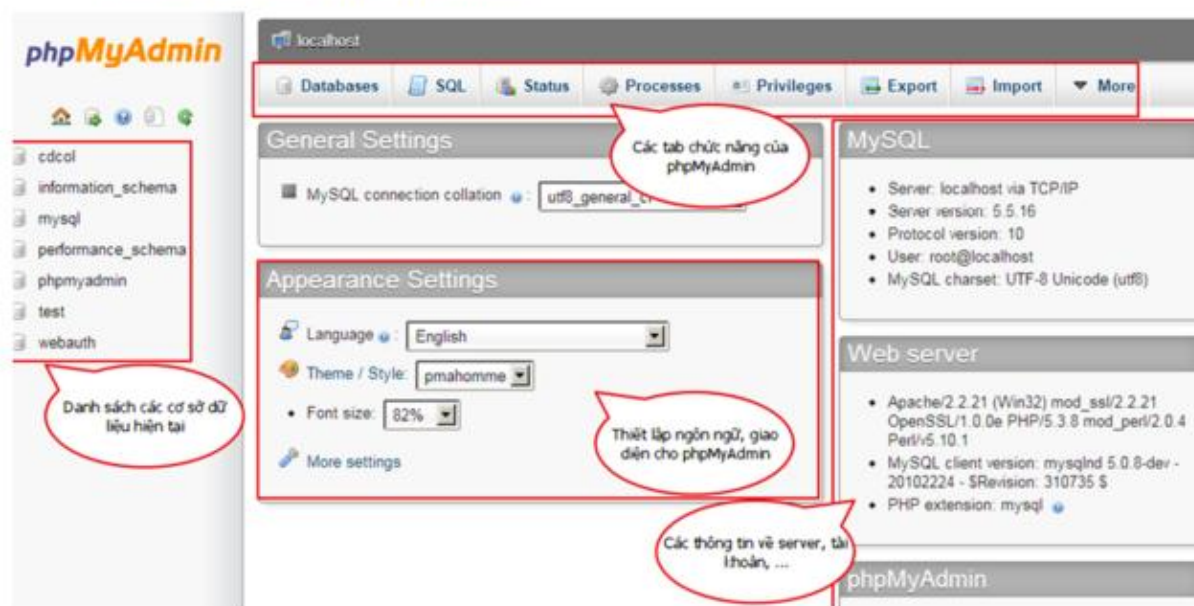
`mysql_error()`: thông báo lỗi (nếu có).

- Bước 5: Đóng kết nối đến server.

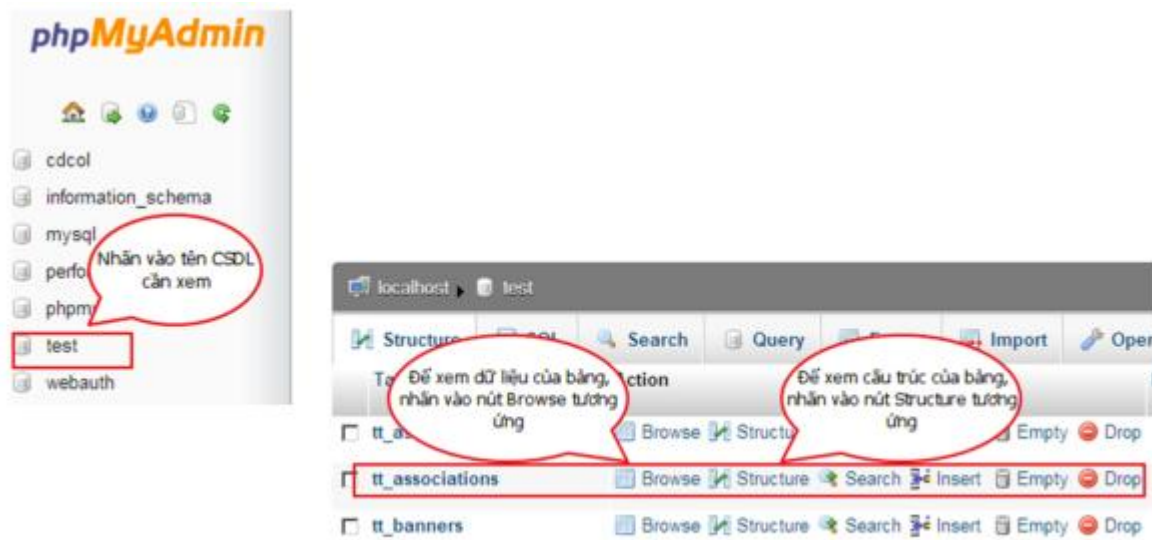
### 3.2. Sử dụng cơ sở dữ liệu MySQL

- 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
<code>MYSQL_ASSOC</code>	<code>\$row["tên cột"]</code>
<code>MYSQL_NUM</code>	<code>\$row[0]</code>
<code>MYSQL_BOTH</code>	<code>\$row[0]</code> hoặc <code>\$row["tên cột"]</code>

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.

- Single Page là toàn bộ nội dung thông tin được trình bày trên một trang web, người dùng chỉ cần kéo chuột để xem mà không cần click hoặc chuyển hướng qua những trang khác. Vì chỉ có trình bày các thông tin trên một trang web nên yêu cầu thiết kế trang này phải bắt buộc thu hút, sáng tạo, đặc biệt phải sử dụng nhiều hình ảnh minh họa cùng ý tưởng độc đáo để thu hút tạo tính tò mò cho người xem kéo xuống xem tiếp các phần sau.

- Master Page là 1 trang web mẫu (template mẫu ) để các trang khác kế thừa nó thông qua 1 đối tượng là ContentPlaceholder...các trang khác khi thiết kế thì nội dung sẽ chỉ nằm trong đối tượng đó. Mục đích chính của Master là nhằm tránh việc design lại giao diện ở mỗi trang mất thời gian cho người thiết kế. Master Page cho phép bố trí layout của ứng dụng một cách phù hợp. Nó có thể nắm giữ các chức năng phổ biến trong toàn bộ website. Vì vậy, chỉ cần bạn có thể xây dựng Master Page và sau đó xây dựng các trang nội dung, và trong khi tạo trang nội dung, bạn ràng buộc nó tới trang master page đã tạo trước đó, và 2 trang sẽ được kết hợp lại khi chạy.

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

- Web Editor là các công cụ dùng để biên tập và chỉnh sửa các nội dung Web phức tạp một cách nhanh chóng và dễ dàng. Hiện nay trên thị trường có rất nhiều công cụ Web Editor khác nhau như: [Codepen](#), [Mozilla Thimble](#), [CSSDesk](#), [StackEdit](#), ...

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

**Có 4 loại kiểu dữ liệu khác nhau có thể sử dụng khi thiết kế trang PHP:**

- **Kiểu Short:** Thẻ mặc định mà các nhà lập trình PHP thường sử dụng.

Ví dụ: `<?php echo "Well come to PHP."; ?>`

- **Kiểu định dạng XML:** Thẻ này có thể sử dụng với văn bản định dạng XML

Ví dụ: `<? php echo "Well come to PHP wicth XML"; ?>`

- **Kiểu Script:** Trong trường hợp bạn sử dụng PHP như một script tương tự khai báo JavaScript hay VBScript:

Ví dụ: `<script language= "php"> echo "Php Script"; </sript>`

- **Kiểu ASP:** Trong trường hợp bạn khai báo thẻ PHP như một phần trong trang ASP.

Ví dụ: `<% echo "PHP - ASP";%>`

### 5.4. Phân trang dữ liệu.

- Phân trang là một phần quan trọng của bất kỳ ngôn ngữ lập trình web động nào, đây cũng là phần gây nhiều khó khăn cho những ai mới làm quen với ngôn ngữ lập trình web. Phân trang trong PHP với một số qui ước để tính toán như sau:

1. \$pages : Tổng số trang
2. \$limit : Số mẫu tin trên một trang
3. \$start : Vị trí mẫu tin bắt đầu trên trang
4. \$curpage : Vị trí trang hiện tại

$\$pages = (\text{tổng số mẫu tin} / \$limit)$  số trang phải được làm tròn lên nếu chia là số lẻ.

$\$start = (\$curpage - 1) * \$limit$

Từ trên dễ dàng suy ra  $\$curpage = (\$start / \$limit) + 1;$

- Dùng vòng lặp for lặp qua tổng số trang để tạo ra các liên kết đến trang và ta sẽ dùng phương thức truyền biến GET để lấy các giá trị về số trang hiện tại hay tổng số trang.

**Cụ thể hóa các vấn đề trên sẽ là đoạn code sau đây:**

```
class Paging{
protected $_total;
protected $_pages;
// Xây dựng hàm tìm tổng số mẫu tin
public function findTotal($db, $table){
 if(isset($_GET['total'])){
 $this->_total = $_GET['total'];
 }else {
 $sql= 'SELECT COUNT(*) FROM '.$table;
 $result = $db->query($sql);
 $row = $db->fetch_array($result);
 $this ->_total = $row[0];
 }
}

// Xây dựng hàm tính số trang
public function findPages($limit){
 $this->_pages = ceil($this->_total / $limit);
}

// Xây dựng hàm tính vị trí mẫu tin bắt đầu từ vị trí trang
function rowStart($limit){
 return (!isset($_GET['page'])) ? 0 : ($_GET['page']-1) * $limit;
}

public function pagesList($curpage){
 $total = $this->_total;
 $pages = $this->_pages;
 if($pages <=1){return "";}
 $page_list="";
 // Tạo liên kết tới trang đầu và trang trang trước
```

```
if($curpage!=1){
 $page_list .= '<a href="'.$_SERVER['PHP_SELF'].'?page=1&total=' .
 $total.'" title="trang đầu">First ';
}

if($curpage > 1){
 $page_list .= '<a href="'.$_SERVER['PHP_SELF'].'?page='.($curpage-
 1).'&total='.$total.'" title="trang trước">< ';
}

// Tạo liên kết tới các trang
for($i=1; $i<=$pages; $i++){
 if($i == $curpage){
 $page_list .= "".$i."";
 }
 else{
 $page_list .= '<a href="'.$_SERVER['PHP_SELF'].'?page='.$i.'
 &total='.$total.'" title="Trang '.$i.">'. $i.';
 }
 $page_list .= " ";
}

// Tạo liên kết tới trang sau và trang cuối
if(($curpage+1)<=$pages){
 $page_list .= '<a href="'.$_SERVER['PHP_SELF'].'?page='.($curpage+1).
 '&total='.$total.'" title="Đến trang sau"> > ';
}

if(($curpage != $pages) && ($pages != 0)){
 $page_list .= '<a href="'.$_SERVER['PHP_SELF'].'?page='.$pages.
 '&total='.$total.'" title="trang cuối"> Last';
}
```

```
 return $page_list;
 } // end pagesList
} // end class
```

**Sau khi tạo xong lớp phân trang bây giờ chúng ta chỉ việc gọi và chạy chương trình.**

```
$paging = new Paging; // khởi tạo lớp phân trang
$limit = 10; // Tổng số mẫu tin
$paging->findTotal($db, 'shops'); // Tổng số trang
$paging->findPages($limit);
$start = $paging->rowStart($limit); // Bắt đầu từ mẫu tin
$page = ($start/$limit)+1; // Trang hiện tại

// Xuất dữ liệu với truy vấn
$sql = 'SELECT * FROM shops LIMIT '.$start.', '.$limit;
// Hiển thị kết quả phân trang
echo $paging->pagesList($page);
```

- Để thử nghiệm sinh viên nhớ tạo csdl nhiều nhiều dữ liệu một chút để test. Bài toán chỉ khó ở giải thuật nhưng với việc xây dựng bằng lớp trong hướng đối tượng việc áp dụng sẽ vô cùng đơn giản. Tất nhiên đây không phải là cách tối ưu nhất, sinh viên có thể cải tiến tùy theo ý thích để phân trang chuyên nghiệp hơn.

- Có thể tìm hiểu thêm về phân trang tại: <http://expressmagazine.net/development/1855/huong-dan-xay-dung-lop-phan-trang-trong-php#sthash.0epZgq66.dpuf>

### 5.5. Lọc và tìm kiếm dữ liệu.

Lọc dùng để xác nhận hoặc kiểm tra xem dữ liệu có đáp ứng đủ điều kiện yêu cầu hay không.

#### 1. Sử dụng hàm Filter để kiểm tra dữ liệu.

- Cú pháp: `filter_var($variable, $filter, $option)` để kiểm tra biến `$variable` với điều kiện đưa vào, biến `$filter` thường mang các giá trị sau đây:

- o `FILTER_VALIDATE_BOOLEAN` // kiểm tra giá trị có phải là kiểu `true/false`.
- o `FILTER_VALIDATE_EMAIL` // kiểm tra giá trị có phải là email hay không
- o `FILTER_VALIDATE_FLOAT` // kiểm tra giá trị có phải là kiểu số thực.
- o `FILTER_VALIDATE_IP` // kiểm tra giá trị có phải là địa chỉ IP.
- o `FILTER_VALIDATE_URL` // kiểm tra xem giá trị có phải là địa chỉ liên kết.
- o ...

### 2. Tự định nghĩa và gọi hàm trong PHP Filter

- Cú pháp: `filter_var($variable, FILTER_CALLBACK, array('options'=> 'myFunc'))` để kiểm tra biến `$variable` với hàm `myFunc()`;

### 3. Sử dụng REGEXP trong PHP Filter

- Cú pháp: hàm `filter_var($variable, FILTER_VALIDATE_REGEXP, $option)` thực hiện các yêu cầu sau:

- o Kiểm tra số điện thoại với định dạng sau 084-08-38.212121
- o Kiểm tra phần mở rộng của FILE có phải là .jpg .gif .png
- o Kiểm tra giá trị nhập vào có phải là số và chữ hay không

4. **Sử dụng hàm `filter_var_array()`:** để kiểm tra, lọc cùng lúc nhiều giá trị.

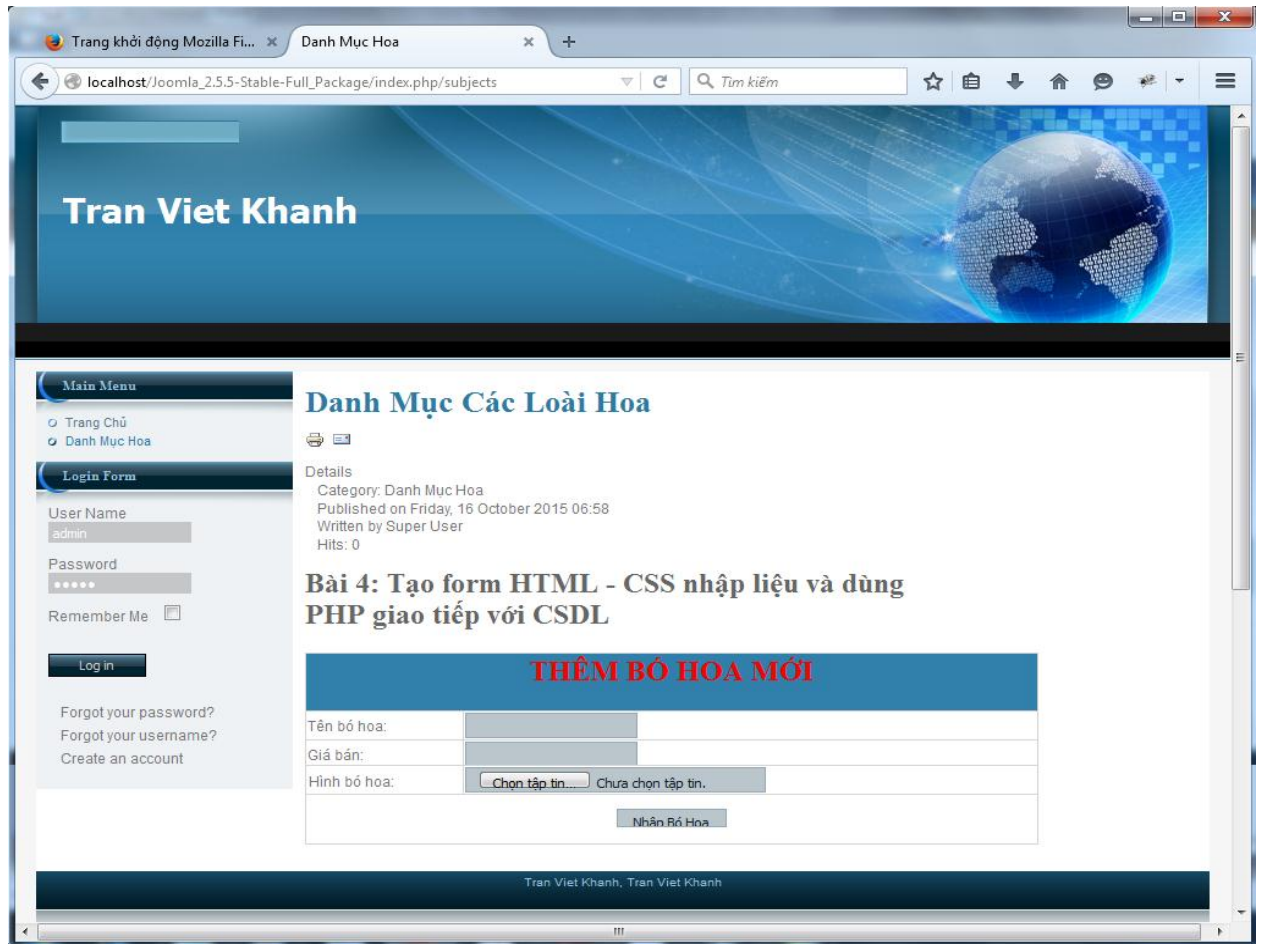
5. **Sử dụng hàm `filter_input(input_type, variable, filter, options)`:** để kiểm tra các giá trị từ FORM. Chức năng này cũng có thể lấy được dữ liệu từ các trường hợp khác như `$_GET`, `$_POST`, `$_COOKIE`, `$_ENV`, `$_SERVER`.

- o `input_type`: kiểu lấy dữ liệu (`INPUT_GET`, `INPUT_POST`, `INPUT_COOKIE`, ...)
- o `variable`: tên của phần tử bạn muốn lấy dữ liệu
- o `filter`: kiểu filter
- o `Options`: mảng tham số

6. **Sử dụng hàm `filter_input_array()`:** tương tự hàm `filter_input` tuy nhiên hàm `filter_input_array()` cho phép thực hiện việc validate và filter cùng lúc trên nhiều giá trị khác nhau.

## B. CÁC DẠNG BÀI TẬP

**BT3-1.** Xây dựng form cho phép người sử dụng thêm thông tin bó hoa vào CSDL và hình ảnh được upload lên thư mục hình\_anh như hình sau đây:



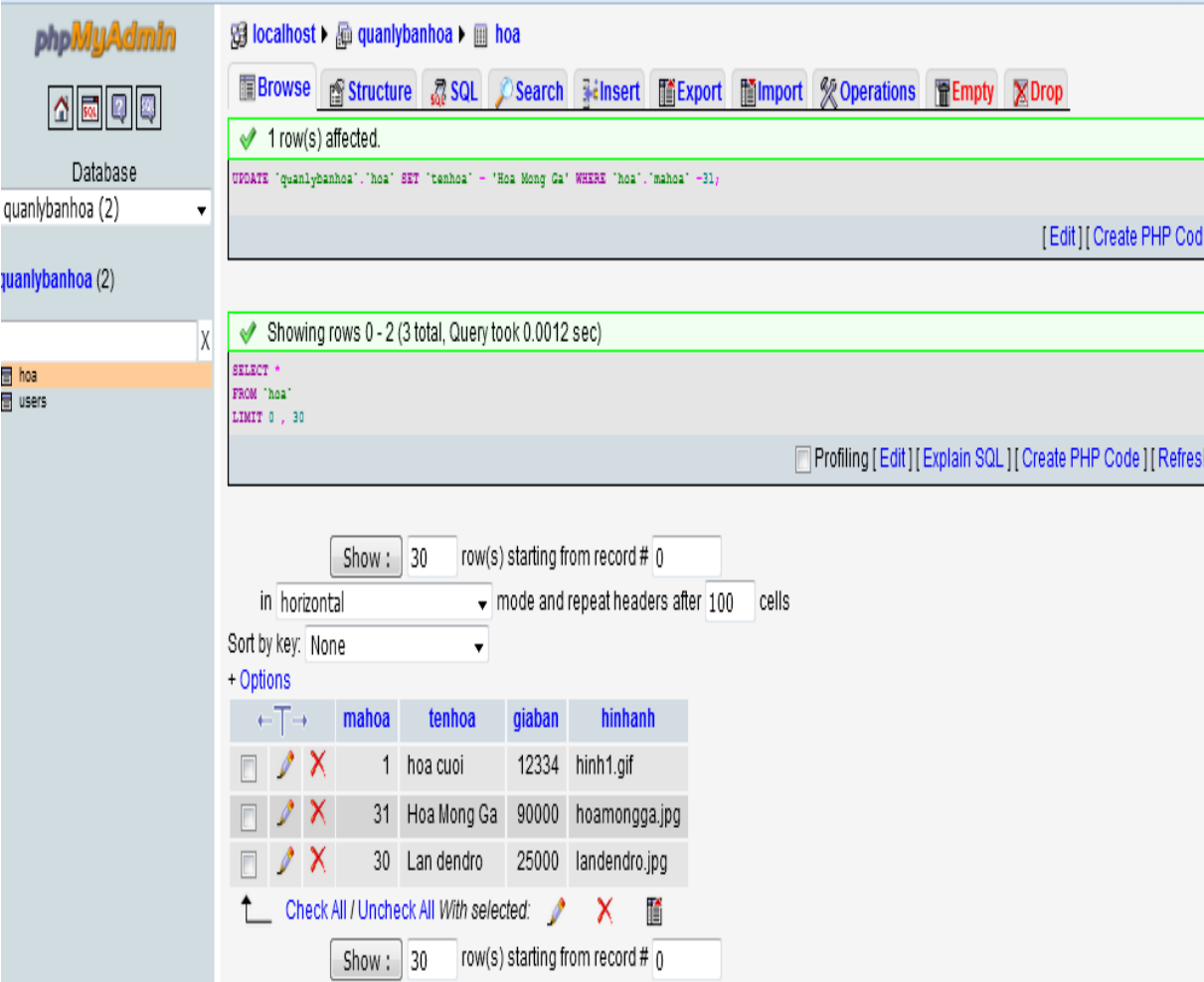
**Hướng dẫn giải bài tập 3-1:** Sinh viên thiết kế form giao diện và viết chương trình theo code mẫu sau đây:

```
<?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:".$_FILES["file"]["error"]."
";
}
else
{
 move_uploaded_file($_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'>Successful !</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">THÊM BÓ HOA MỚI
 </h1></td>
 </tr>
 <tr>
 <td>Tên bó hoa:</td>
 <td><input type="text" name="ten_hoa" /></td>
 </tr>
 <tr>
 <td>Giá bán:</td>
```

```
<td><input type="text" name="gia_ban" /></td>
</tr>
<tr>
<td> Hình bóhoa:</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>
```

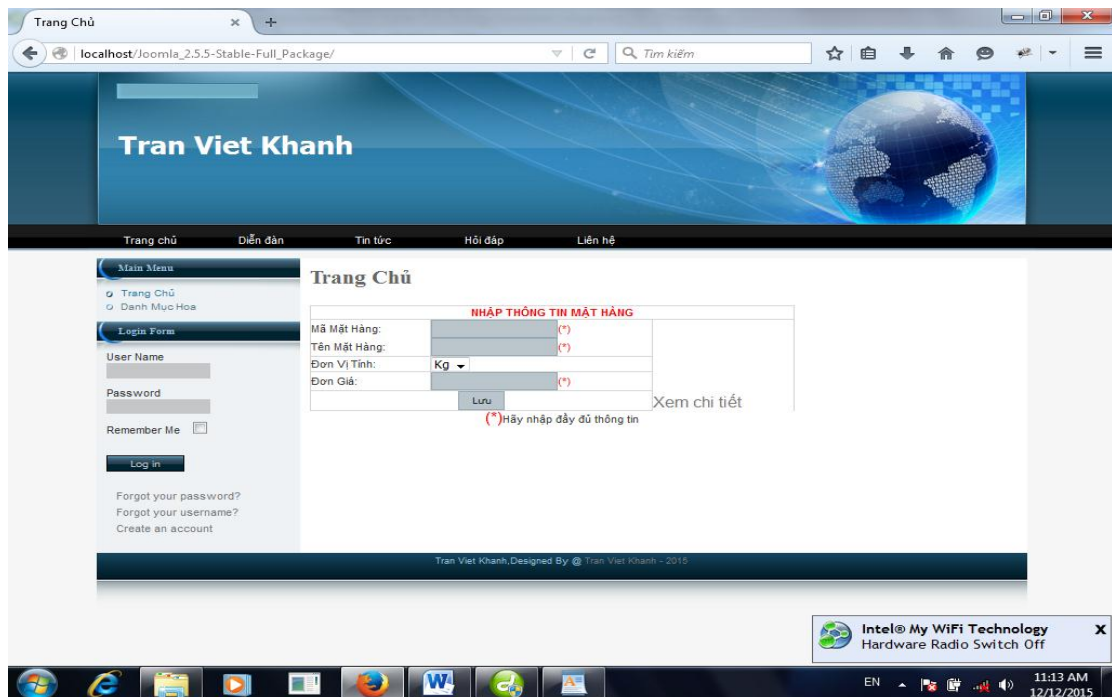
➤ Một số kết quả của bảng hoa trong csdl khi chạy chương trình:



The screenshot shows the phpMyAdmin interface. The left sidebar displays the database structure, including 'quanlybanhoa (2)' and 'hoa'. The main panel shows the 'hoa' table with the following data:

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

**BT3-2.** Xây dựng form cho phép người sử dụng nhập thông tin mặt hàng vào CSDL như hình sau đây:



**Hướng dẫn giải bài tập 3-2:** Sinh viên thiết kế form giao diện và viết chương trình theo code mẫu sau đây:

**<?php**

```
$cn = mysql_connect("localhost","root","") or die("Couldn't to open database");
mysql_select_db("mathang");
if(isset($_POST["MAHH"]) && isset($_POST["TENHH"]) &&
isset($_POST["DVT"]) && isset($_POST["DONGIA"]))
{
 $mahh=$_POST["MAHH"];
 $tenhh=$_POST["TENHH"];
 $dvt=$_POST["DVT"];
 $dongia=$_POST["DONGIA"];
 $sql = "insert into dm_hanghoa values('".$mahh."','".$tenhh."','".$dvt."','".$dongia."','".$null."','".$null."')";
 mysql_query($sql,$cn);
 echo "<div align='center'>Bạn đã lưu thành công</div>";
}
else
{
```

```
 echo "<div align='center'>(*)
 Hãy nhập đầy đủ thông tin</div>";
 }
?>
<form action="<?php echo htmlspecialchars($_SERVER['PHP_SELF']) ?>" method="post"
enctype="multipart/form-data" name="form1" id="form1">
<table width="500" border="1" align="center">
<tr>
<td colspan="3" align="center">NHẬP THÔNG TIN
MẬT HÀNG</td>
</tr>
<tr>
<td>Mã Mật Hàng:</td>
<td><label for="MAHH"></label>
<input type="text" name="MAHH" id="MAHH" />(*)</td>
<td rowspan="8" valign="bottom">Xem chi tiết</td>
</tr>
<tr>
<td>Tên Mật Hàng:</td>
<td><label for="TENHH"></label>
<input type="text" name="TENHH" id="TENHH" />(*)</td>
</tr>
<tr>
<td>Đơn Vị Tính:</td>
<td><select name="DVT"><option>Kg</option>
<option>CAI</option>
</select>
</tr>
<tr>
<td>Đơn Giá:</td>
<td><label for="DONGIA"></label>
<input type="text" name="DONGIA" id="DONGIA" />(*)</td>
</tr>
<tr>
<td colspan="2" align="center"><input type="submit" name="button" id="button"
value="Lưu" style="height:inherit" /></td>
</tr>
</table>
```

</form>

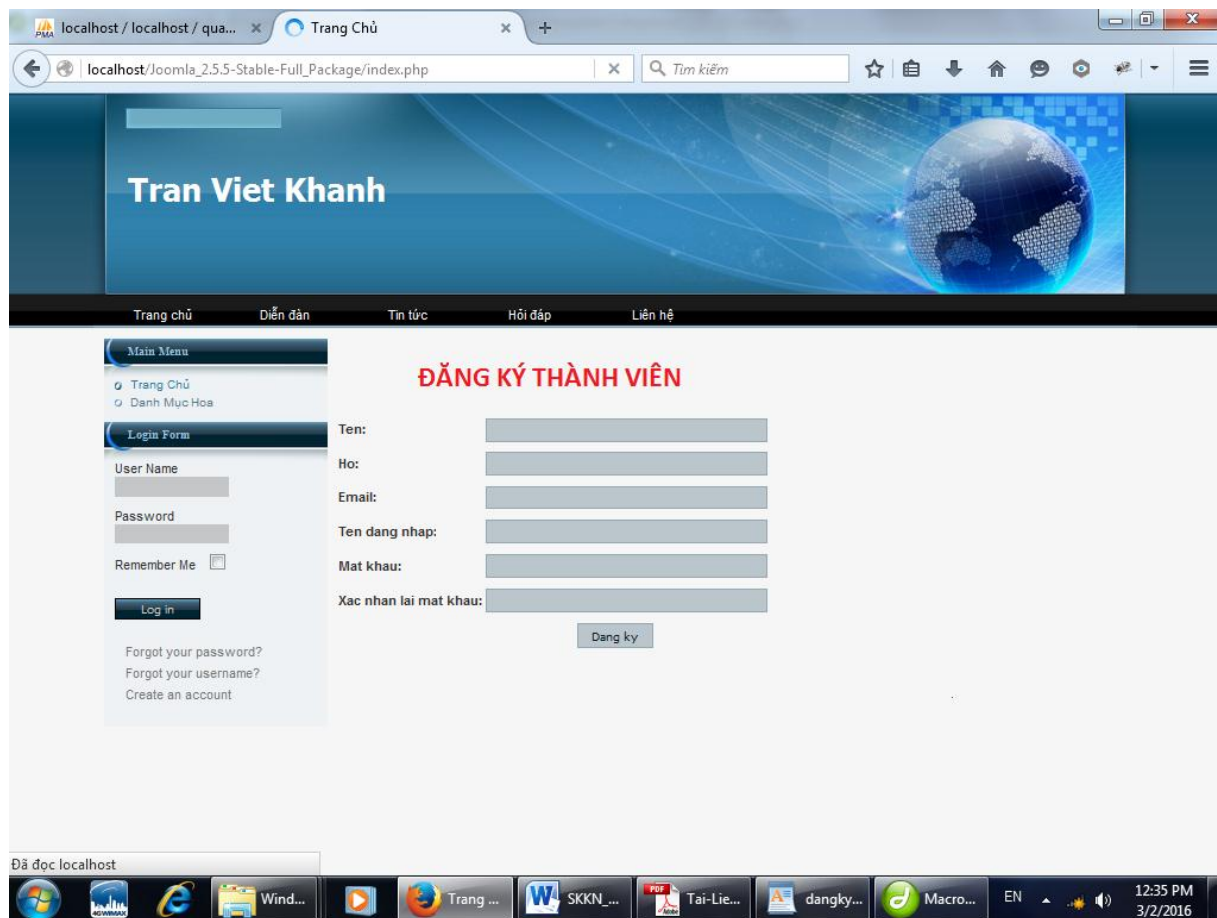
➤ Một số kết quả của bảng dm\_hanghoa trong csdl khi chạy chương trình:

The screenshot shows the phpMyAdmin interface for a database named 'mathang'. The table 'dm\_hanghoa' is selected, and the SQL query 'UPDATE `mathang`.`dm\_hanghoa` SET `DONGIA` = '80000' WHERE `dm\_hanghoa`.`MAHH` = 'mh02';' has been executed, affecting 1 row. The table structure is shown with columns: MAHH, TENHH, DVT, DONGIA, MALOAI. The table data is displayed as follows:

	MAHH	TENHH	DVT	DONGIA	MALOAI
<input type="checkbox"/>	mh01	Tao	Kg	100000	
<input type="checkbox"/>	mh02	Cam	Kg	80000	
<input type="checkbox"/>	mh03	Le	Kg	900000	

Below the table, there are options to check/uncheck all rows with selected. The 'Query results operations' section includes links for 'Print view', 'Print view (with full texts)', 'Export', and 'CREATE VIEW'. A warning message at the bottom states: 'May be approximate. See FAQ 3.11'.

**BT3-3.** Xây dựng form cho phép người sử dụng đăng ký thành viên trên trang web và lưu thông tin thành viên vào CSDL, sinh viên sử dụng hàm ENCODE để mã hóa password trước khi lưu trữ vào cơ sở dữ liệu như hình sau đây:



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

**<?php**

```
$page_title = 'Dang ky';
$scn = mysql_connect("localhost","root","") or die("Couldn't to open database");
mysql_select_db("quanlybanhoa");
if(isset($_POST["submit"]))
{
 $message = NULL;
 if(empty($_POST["first_name"]))
 {
 $fn=FALSE;
 $message .="<p>Ban chua nhap ten</p>";
 }
 else
 {

```

```
 $fn = $_POST["first_name"];
 }
 if(empty($_POST["last_name"]))
 {
 $ln=FALSE;
 $message .="<p>Ban chua nhap ho</p>";
 }
 else
 {
 $ln = $_POST["last_name"];
 }
 if(empty($_POST["email"]))
 {
 $e=FALSE;
 $message .="<p>Ban chua nhap email</p>";
 }
 else
 {
 $e = $_POST["email"];
 }
 if(empty($_POST["username"]))
 {
 $u=FALSE;
 $message .="<p>Ban chua nhap ten dang ky</p>";
 }
 else
 {
 $u = $_POST["username"];
 }
 if(empty($_POST["password1"]))
 {
 $p=FALSE;
 $message .="<p>Ban chua nhap mat khau</p>";
 }
 else
 {
 if($_POST["password1"]==$_POST["password2"])
 {
 $p = $_POST["password1"];
 }
 else
 {
 $p = FALSE;
 }
 }
}
```

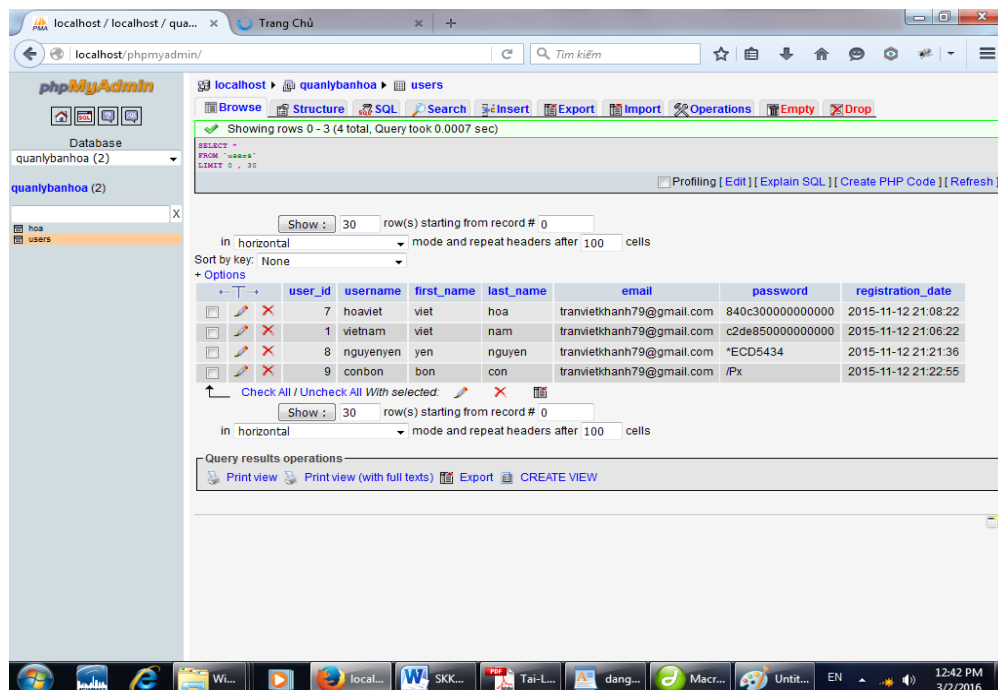
```
 $message .="<p>Mat khau xac nhan khong dung !</p>";
 }
}
if($fn&&$ln&&$e&&$u&&$p)
{
 //$pass = md5("123");
 $query ="insert into
users(username,first_name,last_name,email,password,
registration_date)
values('$u','$fn','$ln','$e',ENCODE($p,'salt'),NOW())";
$result = @mysql_query($query);
if($result){
 echo "<p>Ban da dang ky thanh cong</p>";
 exit();// mysql_close();
}
else
{
 $message = "<p>Ban khong the dang ky. Chung toi xin loi vi su
co nay. ".mysql_error()."</p>";
}
}
if(isset($message))
{
 echo "",$message,"";
}
}
$sql1 = "select username,DECODE(password,'salt') as pwd from users";
$rs=mysql_query($sql1,$cn);
while($row = mysql_fetch_array($rs))
{
 if($row["pwd"]==234) echo "<script
 language='javascript'>location.href='index.php';</script>";
 //echo $row["username"];
}
}
```

?>

```
<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="post">
 <fieldset><legend>Dang Ky Thanh Vien</legend>
 <p>Ten:
 <input type="text" name="first_name" size="50" maxlength="50"
value="<?php if(isset($_POST['first_name'])) echo $_POST['first_name'];?>" />
 </p>
 <p>Ho:
```

```
<input type="text" name="last_name" size="50" maxlength="50"
value="<?php if(isset($_POST['last_name'])) echo $_POST['last_name'];?>" />
</p>
<p>Email:
<input type="text" name="email" size="50" maxlength="50"
value="<?php if(isset($_POST['email'])) echo $_POST['email'];?>" />
</p>
<p>Ten dang nhap:
<input type="text" name="username" size="50" maxlength="50"
value="<?php if(isset($_POST['username'])) echo $_POST['username'];?>" />
</p>
<p>Mat khau:
<input type="password" name="password1" size="50" maxlength="50"/>
</p>
<p>Xac nhan lai mat khau:
<input type="password" name="password2" size="50" maxlength="50"/>
</p>
</fieldset>
<div align="center"><input type="submit" name="submit" value="Dang ky"
style="height:inherit" /> </div>
</form>
```

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



	user_id	username	first_name	last_name	email	password	registration_date
<input type="checkbox"/>	7	hoaviet	viet	hoa	tranvietkhanh79@gmail.com	840c300000000000	2015-11-12 21:08:22
<input type="checkbox"/>	1	vietnam	viet	nam	tranvietkhanh79@gmail.com	c2de850000000000	2015-11-12 21:06:22
<input type="checkbox"/>	8	nguyenyen	yen	nguyen	tranvietkhanh79@gmail.com	*ECD5434	2015-11-12 21:21:36
<input type="checkbox"/>	9	conbon	bon	con	tranvietkhanh79@gmail.com	/Px	2015-11-12 21:22:55

**BT4-1.** Xây dựng form cho phép người sử dụng đăng nhập vào trang web, tạo và gọi hàm `checkInput` kiểm tra tính hợp lệ của dữ liệu khi người dùng nhập dữ liệu trên form (nếu tên đăng nhập là “vietkhanh” và mật khẩu là “12345” thì gọi hiển thị trang chủ, nếu không hợp lệ thì báo username và password không đúng) như hình sau đây:



**Hướng dẫn giải bài tập 4-1:** 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']); // gọi hàm checkInput
 $password = checkInput($_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
($_SERVER['PHP_SELF']) ?>" method="POST">
 <fieldset>
 <legend><h2>NHẬP THÔNG TIN ĐĂNG NHẬP</h2></legend>
 <p>Tên đăng nhập:
 <input type="text" name="username" size="15" maxlength="25"
 value="<?php if(isset($_POST['username']))echo $username;?>"
 </p>
 <p>
 Mật khẩu:<input type="password" name="password"
 size="22" maxlength="20" value="<?php
 if(isset($_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	<input type="checkbox"/> vietkhanh	12345	Hiện thị trangchu.php
test 2	\vietkhanh	\12345	Hiện thị trangchu.php
test 3	Viet khanh	12345	Hiện thi thông báo username hoặc password không đúng
Test 4	vietkhanh	12345	Hiện thị trangchu.php

**BT5-1.** Xây dựng form cho phép người sử dụng tìm kiếm thông tin người dùng như hình sau đây:



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

## 1. Xây dựng Cơ sở dữ liệu để tìm kiếm

- Trước tiên sinh viên tạo một database tên là basic, sau đó sử dụng đoạn code dưới đây để tạo table và thêm một số dòng dữ liệu.

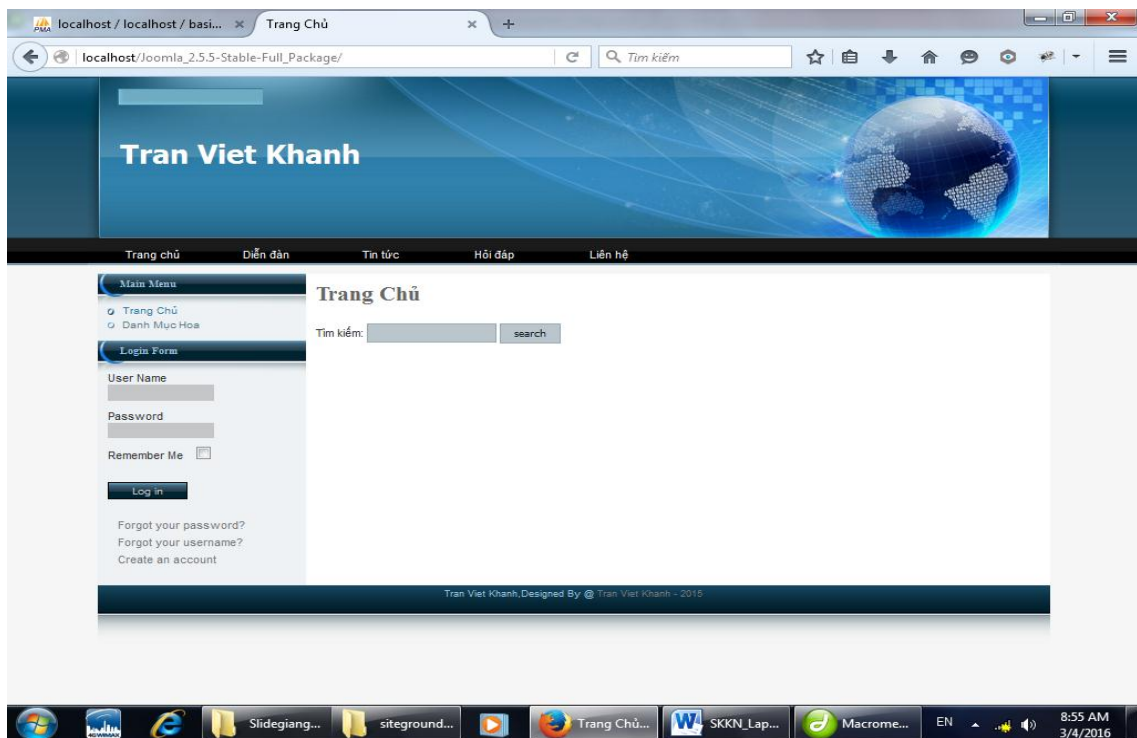
```
CREATE DATABASE `basic`
CREATE TABLE `users` (
 `user_id` int(11) unsigned NOT NULL AUTO_INCREMENT,
 `username` varchar(50) COLLATE utf8_unicode_ci DEFAULT NULL,
 `password` char(32) COLLATE utf8_unicode_ci DEFAULT NULL,
 `email` varchar(100) COLLATE utf8_unicode_ci DEFAULT NULL,
 `address` varchar(100) COLLATE utf8_unicode_ci DEFAULT NULL,
 PRIMARY KEY (`user_id`)
```

```
) ENGINE=InnoDB AUTO_INCREMENT=11 DEFAULT CHARSET=utf8
COLLATE=utf8_unicode_ci;

/* Nhập dữ liệu cho bảng `users` */

insert into `users`(`user_id`,`username`,`password`,`email`,`address`) values
(1,'kaito','456','kaito@freetuts.net','House
USA'),(2,'thefafheart','456','thefafheart@freetuts.net','House
USA'),(3,'superkaito','456','superkaito@gmail.com','House Viet
Nam'),(4,'kaitosolo','1789','kaito@yahô.com','abc/fdc'),(5,'van
canh','798','vancanh@freetuts.net','Bien
hoa'),(6,'yongc','789','yongc@freetuts.net','Bien
Hoa'),(7,'hafheart789','789','hafheart789@yahoo.com','Ha
tinh'),(8,'yongc456','56465','yongc789@gmail.com','Bien Hoa 2'),(9,'van
canh456','564564','vancanh@gmail.com','Bien Hoa
3'),(10,'kaitocode','87897','kaitocode@yahoo.com','TPHCM');
```

## 2. Xây dựng Form tìm kiếm như hình



### 3. Check submit form tìm kiếm & kết nối Database

- **Kiểm tra submit FORM**

- Trước tiên chúng ta cần kiểm tra người dùng có nhấn vào button tìm kiếm hay không? Bắt buộc chúng ta phải thực hiện bước này tại vì khi submit form mới cần xử lý truy vấn database để lấy thông tin tìm kiếm.

- Bình thường việc check Submit form ta sẽ dùng `$_POST` vì chúng ta sử dụng phương thức GET nên ta phải sử dụng biến `$_GET` để kiểm tra. Nhưng trong bài này sinh viên sẽ thay thế `$_GET` bằng một biến môi trường khác chính là `$_REQUEST` có tác dụng là nhận request từ cả 2 phương thức trên.

```
if (isset($_REQUEST['ok'])) {
 $search = addslashes($_GET['search']);
 if (empty($search)) {
 echo "Nhập dữ liệu cần tìm kiếm";
 } else {
 // Phan dung vong lap while show du lieu
 }
}
```

- `$_REQUEST['ok']` chính là tên của button tìm kiếm. Như vậy để kiểm tra người dùng click nút tìm kiếm hay chưa thì ta chỉ cần kiểm tra tên của button đó có tồn tại hay không.

- **Kết nối Database**

Tiếp theo chúng ta phải thao tác với database vì vậy cần phải thực hiện việc kết nối như sau.

```
if (isset($_REQUEST['ok'])) {
 // Kết nối sql
 mysql_connect("localhost", "root", "") or die ("Couldn't to open database");
 // Chọn database
 mysql_select_db("basic");
}
```

### 4. Xử lý tìm kiếm với lệnh Like trong MySql

- Lệnh LIKE trong MySQL, đây là một lệnh dùng để tìm kiếm dữ liệu ở mức độ gần giống, để rõ hơn thì sinh viên xem bài lệnh LIKE trong MySQL.
- Tất cả những đoạn code kết nối cơ sở dữ liệu và thực hiện truy vấn tìm kiếm dữ liệu đều đặt trong đoạn code kiểm tra submit form. Như vậy ta có fullcode cho phần PHP như sau:

```
if (isset($_REQUEST['ok'])) {
 // Gán hàm addslashes để chống sql injection
 $search = addslashes($_GET['search']);
 // Dùng lệnh like và toán tử % của php để tìm kiếm dữ liệu chính xác hơn.
 $query = "select * from users where username like '%$search%'";
 // Kết nối sql
 mysql_connect("localhost", "root", "") or die ("Couldn't to open database");
 // Chọn database
 mysql_select_db("basic");
 // Thực thi câu truy vấn
 $sql = mysql_query($query);
 // Đếm số dòng trả về trong sql.
 $num = mysql_num_rows($sql);
 // Nếu $search rỗng thì báo lỗi người dùng chưa nhập liệu mà đã nhấn submit.
 if (empty($search)) {
 echo "Nhập dữ liệu cần tìm kiếm";
 } else {
 /* Ngược lại nếu người dùng nhập liệu thì tiến hành xử lý show dữ liệu.
 Nếu $num > 0 hoặc $search khác rỗng tức là có dữ liệu show ra màn hình,
 ngược lại thì báo lỗi. */
 if ($num > 0 && $search != "") {
 // Dùng $num để đếm số dòng trả về.
 echo "$num ket qua tra ve voi tu khoa $search";
 }
 }
}
```

```
/* Vòng lặp while & mysql_fetch_assoc dùng để lấy toàn bộ dữ liệu có
trong table và trả về dữ liệu ở dạng array. */
while ($row = mysql_fetch_assoc($sql)) {
 $id = $row['user_id'];
 $username = $row['username'];
 $password = $row['password'];
 $email = $row['email'];
 $address = $row['address'];
 echo "
<h3>Ho ten: $username (id: $id)</h3>

 Mat Khau: $password
Email: $email

 Dia chi: $address
";
}
} else {
 echo "Khong tim thay ket qua!";
}
```

### **FullCode của toàn bộ chương trình:**

**<?php**

```
if (isset($_REQUEST['ok']))
{
 // Gán hàm addslashes để chống sql injection
 $search = addslashes($_GET['search']);
 // Dùng lệnh like và toán tử % của php để tìm kiếm dữ liệu chính xác hơn.
 $query = "select * from users where username like '%$search%'";
 // Kết nối sql
 mysql_connect("localhost", "root", "") or die ("Couldn't to open database");
 // Chọn database
 mysql_select_db("basic");
 // Thực thi câu truy vấn
 $sql = mysql_query($query);
```

```
// Đếm số dòng trả về trong sql.
$num = mysql_num_rows($sql);
// Nếu $search rỗng thì báo lỗi người dùng chưa nhập liệu mà đã nhấn submit.
if (empty($search)) {
 echo "Nhập dữ liệu cần tìm kiếm";
}
else
{
 /* Ngược lại nếu người dùng nhập liệu thì tiến hành xử lý show dữ liệu.
 Nếu $num > 0 hoặc $search khác rỗng tức là có dữ liệu show ra màn hình,
 ngược lại thì báo lỗi. */
 if ($num > 0 && $search != "")
 {
 // Dùng $num để đếm số dòng trả về.
 echo "$num ket qua tra ve voi tu khoa $search";
 /* Vòng lặp while & mysql_fetch_assoc dùng để lấy toàn bộ dữ liệu có
 trong table và trả về dữ liệu ở dạng array. */
 while ($row = mysql_fetch_assoc($sql)) {
 $id = $row['user_id'];
 $username = $row['username'];
 $password = $row['password'];
 $email = $row['email'];
 $address = $row['address'];
 echo "
<h3>Họ tên: $username (id: $id)</h3>

 Mật Khẩu: $password
Email: $email

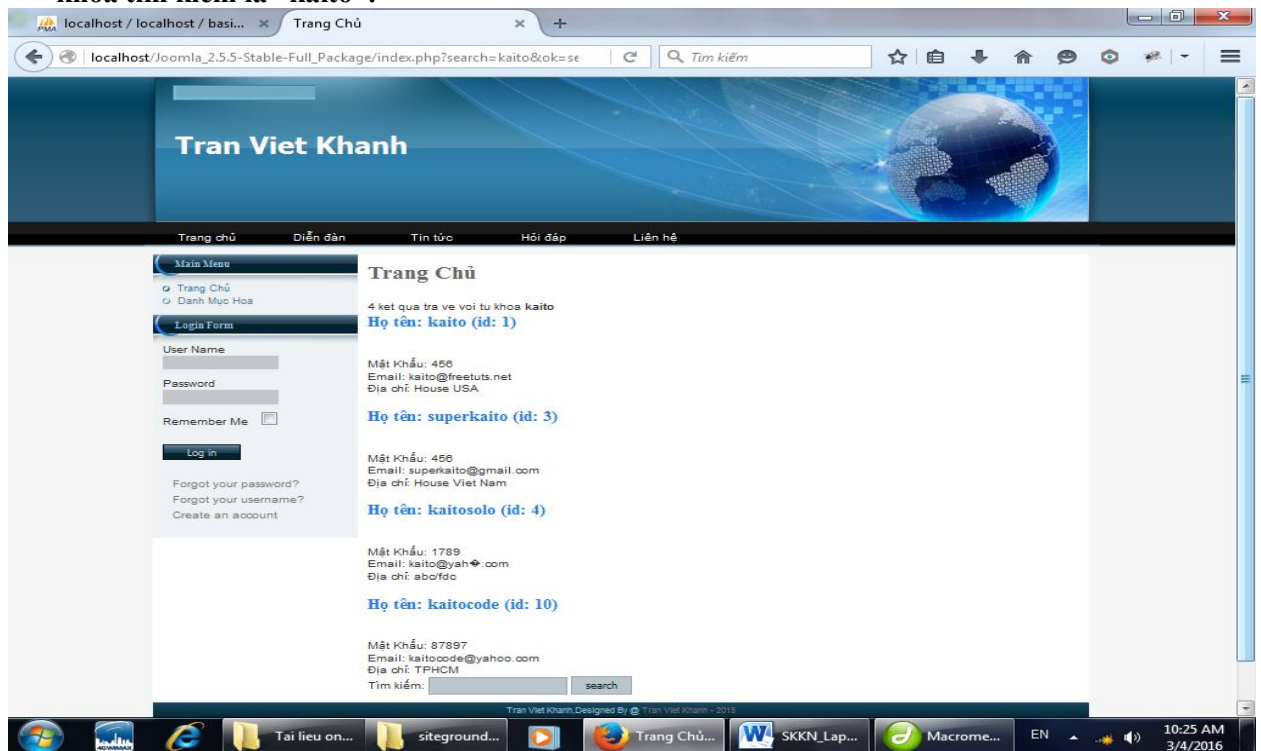
 Địa chỉ: $address
";
 }
 }
 else
```

```
{
 echo "Khong tim thay ket qua!";
}
}
```

?>

```
<form action="<?php echo htmlspecialchars($_SERVER['PHP_SELF']) ?>"
method="get"> Tìm kiếm: <input type="text" name="search" />
 <input type="submit" name="ok" value="search" style="height:inherit" />
</form>
```

- Một số kết quả tìm kiếm trong csdl hiển thị trên màn hình khi chạy chương trình với từ khóa tìm kiếm là “kaito”:



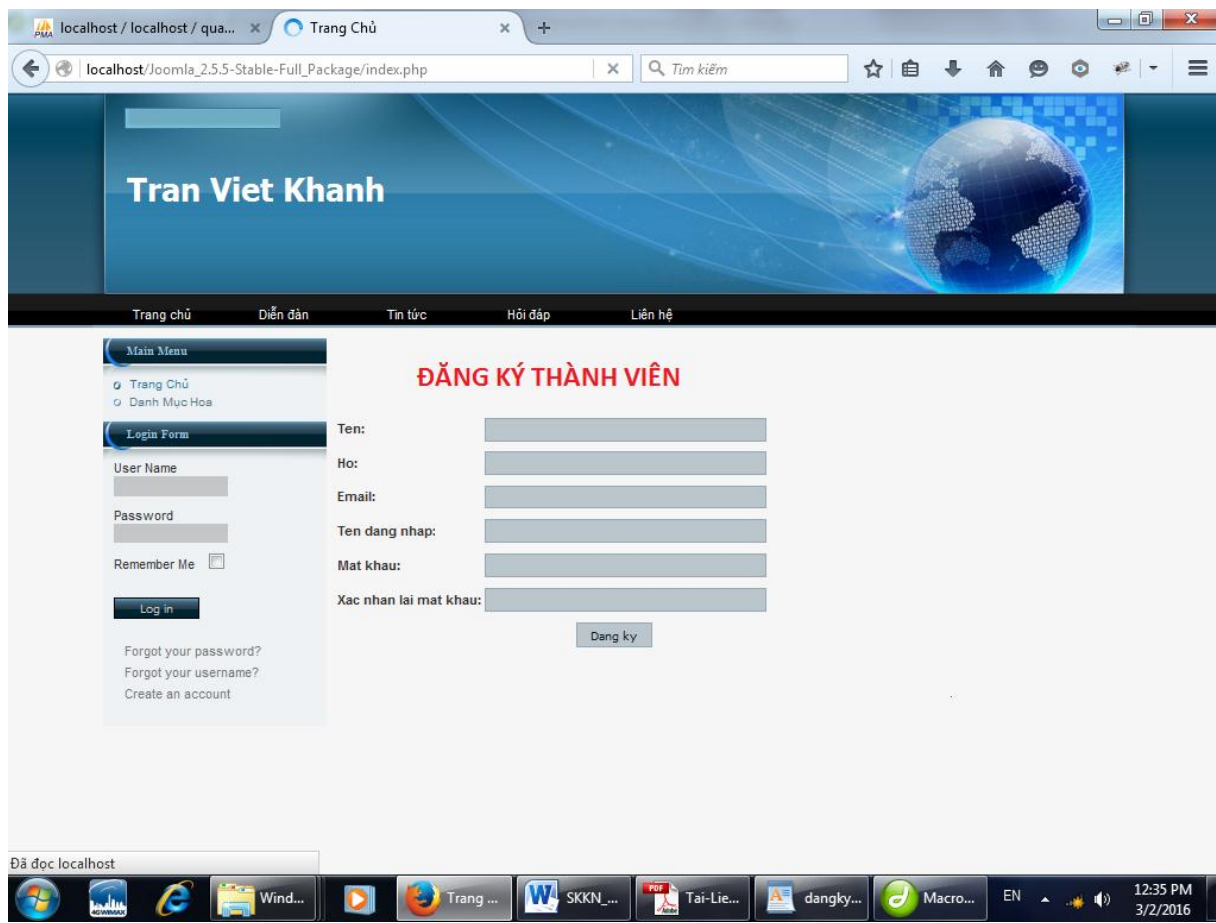
## C.BÀI TẬP NÂNG CAO

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

**BT3-4.** Xây dựng form cho phép người sử dụng đăng ký thành viên trên trang web và lưu

## Truy Xuất CSDL MySQL Với PHP

thông tin thành viên vào CSDL, khác với **BT3-3** sinh viên sử dụng hàm **PASSWORD** hoặc **MD5** để mã hóa password trước khi lưu trữ vào CSDL như hình sau đây:



**Gợi ý làm bài:** Thiết kế form giao diện như **BT3-3**:

- Mã hóa password bằng hàm **PASSWORD** hoặc **MD5**.
- Nhận xét kết quả của mật khẩu với **BT3-3**.

**BT4-2.** Xây dựng WebForm cho phép người sử dụng đăng nhập vào trang web, viết hàm kiểm tra người dùng có tồn tại trong CSDL hay không? Nếu có thì gọi hiển thị trang chủ, nếu không thì thông báo phải đăng ký thành viên như hình sau đây:

# Truy Xuất CSDL MySQL Với PHP



**Gợi ý làm bài:** Thiết kế form giao diện như **BT4-1**:

- Viết hàm kiểm tra người dùng có phải là thành viên của trang Web hay không. Nếu có hiển thị trang chủ. Nếu không thông báo cho người dùng phải đăng ký thành viên.
- Nhận xét kết quả với **BT4-1**.

**BT5-2.** Xây dựng form cho phép người sử dụng tìm kiếm thông tin theo user\_id, email, address,... như hình sau đây:

# Truy Xuất CSDL MySQL Với PHP

---



**Gợi ý làm bài:** Thiết kế form giao diện như **BT5-1**:

- Viết hàm tìm kiếm thông tin theo các tiêu chí (user\_id, address,...).
- Nhận xét kết quả với **BT5-1**.

### Chương 4

## QUẢN TRỊ WEBSITE

---

### A. TÓM TẮT LÝ THUYẾT

#### §6 : PHÂN QUYỀN VÀ QUẢN TRỊ NGƯỜI DÙNG

##### 6.1 Phân quyền người dùng.

Trong website chúng ta phân ra làm 6 nhóm users như sau:

- **Super Admin** : Là đẳng tối cao và là Group có quyền hạn cao nhất.
- **Administrator**: Là user thuộc cấp dưới của Group Super Admin, cũng có quyền hạn cao nhất, và chỉ bị hạn chế chức năng là không xóa được tài khoản thuộc Group Super Admin.
- **Editor**: Là trợ lý của Super Admin & Administrator, có nhiệm vụ duyệt bài viết và xuất bản các bài viết của các tác giả khác, nó chỉ có duy nhất chức năng quản lý bài viết mà thôi.
- **Author**: Là tác giả và chỉ có quyền quản lý toàn bộ bài viết do họ viết ra.
- **Contributor**: Là user thành viên của website, nói chính xác là dân viết thuê, họ biên tập bài viết xong thì chỉ có thể edit chứ không được cấp quyền hạn xuất bản bài viết đó.
- **Subscriber**: Là user vô danh (không là thành viên của website) chỉ có nhiệm vụ dạo quanh website từ bên ngoài và không làm được gì khác.

##### 6.2 Quản trị người dùng và thành viên.

Để có một website chạy trên mạng thì cần phải có:

**Web hosting**: nơi đặt nội dung Website. Nếu truy cập Internet thông thường qua các nhà cung cấp dịch vụ Internet (ISP) thì địa chỉ IP của máy chúng ta luôn bị thay đổi. Do đó máy khác không thể truy cập dữ liệu của máy chúng ta qua Internet được. Nếu đặt nội dung Website của chúng ta trên máy chủ chuyên dụng thì địa chỉ IP của máy đó là cố định các máy khác dễ dàng truy cập Website của chúng ta qua Internet.

**Domain:** định danh của Website trên Internet. Domain thực chất là địa chỉ IP được biến đổi thông qua DNS. VD: 222.135.35.15 – [www.abc.com](http://www.abc.com). DNS Server: là server có nhiệm vụ ánh xạ địa chỉ IP thành domain và ngược lại.

Khi một website được đưa lên mạng Internet thì cần phải quản trị người dùng và thành viên nhằm đạt các mục tiêu sau đây:

- Cho phép tạo và thêm người dùng mới vào một trang web.
- Xem danh sách của người dùng và thay đổi chi tiết về người dùng.
- Thay đổi cấp độ và phân quyền cho người dùng.
- Bảo mật và an toàn dữ liệu cho trang Web.

### 6.3 Sử dụng và tích hợp công cụ Upload File quản trị cập nhật nội dung web.

Có rất nhiều công cụ dùng để quản trị nội dung Website, và cPanel là một trong những công cụ dùng để quản trị cập nhật nội dung Web. Một số tính năng của cPanel như sau:

Quản lý file: cpanel-files

Quản lý email: cpanel-mail

Bảo mật: cpanel-security

Quản lý domain: cpanel-domain

Quản lý cơ sở dữ liệu: cpanel-database

Quản lý software và services: cpanel-servicer-software

Quản lý nâng cao: cpanel-advanced

Thống kê và logs: cpanel-logs

## §7 : GIỎ HÀNG, SESSION, COOKIE

## 7.1 Xây dựng giỏ hàng.

- Tạo ứng dụng giỏ hàng hay Shopping Cart luôn luôn là một phần quan trọng nhất đối với người làm web. Với sự phát triển như vũ bão của thương mại điện tử, thì ngày càng có nhiều website bán hàng với đủ thể loại xuất hiện. Thực tế thì khi xây dựng một website bán hàng, các bạn không cần phải tự lập trình gì nhiều, vì có rất nhiều mã nguồn mở hỗ trợ việc xây dựng một cửa hàng online như Magento, CS Cart, Presstashop..... Nhưng làm gì thì làm, nếu các bạn có thể tự xây dựng cho mình một giỏ hàng riêng, thì vừa có thể nâng cao khả năng lập trình, vừa có thể hiểu rõ hơn về cơ chế mà một website bán hàng cần có.

- Đầu tiên, sinh viên cần tạo 4 file sau đây cho giỏ hàng (shopping cart) của mình :

1. **Config.php** (kết nối cơ sở dữ liệu)
2. **Index.php** (trang hiển thị danh sách sản phẩm từ database)
3. **Cart\_update.php** (cập nhật giỏ hàng)
4. **View\_cart.php** (xem tổng quan giỏ hàng trước khi thanh toán)

Sinh viên tiến hành từng bước như sau:

### Bước 1 : Tạo cơ sở dữ liệu

Trước hết, chúng ta cần có một dữ liệu lưu thông tin sản phẩm như tên sản phẩm, hình ảnh sản phẩm đó... như đoạn sql sau :

```
CREATE TABLE IF NOT EXISTS `products` (
 `id` int(11) NOT NULL AUTO_INCREMENT,
 `product_code` varchar(60) NOT NULL,
 `product_name` varchar(60) NOT NULL,
 `product_desc` tinytext NOT NULL,
 `product_img_name` varchar(60) NOT NULL,
 `price` decimal(10,2) NOT NULL,
 PRIMARY KEY (`id`),
 UNIQUE KEY `product_code` (`product_code`)
) AUTO_INCREMENT=1 ;
```

- Nếu sinh viên không biết dùng câu lệnh sql để tạo, thì có thể vào **PhpMyAdmin** để tạo theo cấu trúc sql bên trên. Khi tạo bảng (table) xong, các bạn có thể chèn thêm danh sách các sản phẩm vào table như hình minh họa sau :

id	product_code	product_name	product_desc	product_img_name	price
1	PD1001	Android Phone	----	product_image.jpg	200
2	PD1002	Television DX	----	product_image2.jpg	500
3	PD1003	Wrist watch	----	product_image3.jpg	400

## Bước 2 : kết nối cơ sở dữ liệu

- Một khi chúng ta đã có dữ liệu lưu trữ thông tin sản phẩm, chúng ta sẽ tiến hành kết nối với dữ liệu để hiển thị sản phẩm ra màn hình. Sinh viên mở file **config.php** ra và copy đoạn code sau vào bên trong nó :

```
<?php
$currency = '$'; // đơn vị tiền tệ
$db_username = 'root'; // user name
$db_password = ''; // password
$db_name = 'demo'; // tên cơ sở dữ liệu
$db_host = 'localhost';
$mysqli = new mysqli($db_host, $db_username, $db_password, $db_name);
?>
```

Các bạn nhớ thay đổi giá trị theo các ghi chú mà mình ghi bên cạnh nhé.

## Bước 3 : Hiển thị thông tin sản phẩm và giỏ hàng

- Bây giờ là lúc mà chúng ta sẽ hiển thị danh sách các sản phẩm cần bán,. Vì chúng ta sẽ sử dụng session để lưu tạm thông tin giỏ hàng, nên việc đầu tiên cần làm là chúng ta phải kích hoạt session . Mở file **index.php** ra và copy đoạn code sau dán vào nó :

```
<?php
session_start();
include_once("config.php");
?>
```

- Trang **index.php** này sẽ hiển thị 2 nội dung, nội dung thứ 1 sẽ liệt kê các sản phẩm (nằm ở giữa) và nội dung thứ 2 hiển thị giỏ hàng (nằm bên phải kế bên).

- **Liệt kê sản phẩm**

- Sinh viên copy đoạn code sau và dán nó vào tiếp trong file index.php để hiển thị các sản phẩm từ database :

```
<div class="products">
<?php
//current URL of the Page. cart_update.php redirects back to this URL
$current_url =
base64_encode("http://".$_SERVER['HTTP_HOST'].$_SERVER['REQUEST_URI']);

$results = $mysqli->query("SELECT * FROM products ORDER BY id ASC");
if ($results) {
 //output results from database
 while($obj = $results->fetch_object())
```

```
{
 echo '<div class="product">';
 echo '<form method="post" action="cart_update.php">';
 echo '<div class="product-thumb"></div>';
 echo '<div class="product-content"><h3>'. $obj->product_name.'</h3>';
 echo '<div class="product-desc">'. $obj->product_desc.'</div>';
 echo '<div class="product-info">Price '. $currency. $obj->price.'
<button class="add_to_cart">Add To Cart</button></div>';
 echo '</div>';
 echo '<input type="hidden" name="product_code" value="'. $obj->product_code.'" />';
 echo '<input type="hidden" name="type" value="add" />';
 echo '<input type="hidden" name="return_url"
value="'. $current_url.'" />';
 echo '</form>';
 echo '</div>';
}

?>
</div>
```

- **Thông tin giỏ hàng :**

- Sau đó copy đoạn code sau và dán tiếp vào file **index.php** để hiển thị giỏ hàng :

```
<div class="shopping-cart">
<h2>Your Shopping Cart</h2>
<?php
if(isset($_SESSION["products"]))
{
 $total = 0;
 echo '';
 foreach ($_SESSION["products"] as $cart_itm)
 {
 echo '<li class="cart-itm">';
 echo '<a
href="cart_update.php?removep='.$cart_itm["code"].'&return_url='.$current_u
rl.'">×';
 echo '<h3>'. $cart_itm["name"].'</h3>';
 echo '<div class="p-code">P code : '. $cart_itm["code"].'</div>';
 echo '<div class="p-qty">Qty : '. $cart_itm["qty"].'</div>';
 echo '<div class="p-price">Price
:'. $currency. $cart_itm["price"].'</div>';
 echo '';
 $subtotal = ($cart_itm["price"]*$cart_itm["qty"]);
 $total = ($total + $subtotal);
 }
}
```

```
 }
 echo '';
 echo 'Total :
'. $currency. $total. ' Check-
out!';
 echo 'Empty
Cart';
} else {
 echo 'Your Cart is empty';
}
?>
</div>
```

### Bước 4 : Cập nhật giỏ hàng

- Giỏ hàng sẽ có chức năng là thêm và xóa sản phẩm khi đưa vào giỏ. Khi click vào nút **“Add to Cart”**, form sẽ gửi một giá trị hidden là mã sản phẩm (product code) với số lượng (quantity) đến file **Cart\_update.php** thông qua phương thức **POST**. Chúng ta sẽ sử dụng những thông tin được gửi đến này để lấy thông tin sản phẩm từ database. Sinh viên copy đoạn code bên dưới và dán vào file **Cart\_update.php**. Mỗi một câu lệnh có để ghi chú, sinh viên có thể xem những dòng ghi chú này để hiểu rõ code hơn.

```
<?php
session_start(); //start session
include_once("config.php"); //include config file

//empty cart by distroying current session
if(isset($_GET["emptycart"]) && $_GET["emptycart"]==1)
{
 $return_url = base64_decode($_GET["return_url"]); //return url
 session_destroy();
 header('Location: '.$return_url);
}

//add item in shopping cart
if(isset($_POST["type"]) && $_POST["type"]=="add")
{
 $product_code = filter_var($_POST["product_code"],
FILTER_SANITIZE_STRING); //product code
 $product_qty = filter_var($_POST["product_qty"],
FILTER_SANITIZE_NUMBER_INT); //product code
 $return_url = base64_decode($_POST["return_url"]); //return url

 //limit quantity for single product
 if($product_qty > 10) {
```

```
 die('<div align="center">This demo does not allowed more than 10
quantity!
<a href="http://sanwebe.com/assets/paypal-shopping-cart-
integration/">Back To Products.</div>');
 }

 //MySql query - get details of item from db using product code
 $results = $mysqli->query("SELECT product_name,price FROM products WHERE
product_code='$product_code' LIMIT 1");
 $obj = $results->fetch_object();

 if ($results) { //we have the product info

 //prepare array for the session variable
 $new_product = array(array('name'=>$obj->product_name,
'code'=>$product_code, 'qty'=>$product_qty, 'price'=>$obj->price));

 if(isset($_SESSION["products"])) //if we have the session
 {
 $found = false; //set found item to false

 foreach ($_SESSION["products"] as $cart_itm) //loop through
session array
 {
 if($cart_itm["code"] == $product_code){ //the item exist in
array

 $product[] = array('name'=>$cart_itm["name"],
'code'=>$cart_itm["code"], 'qty'=>$product_qty,
'price'=>$cart_itm["price"]);
 $found = true;
 }else{
 //item doesn't exist in the list, just retrieve old info
and prepare array for session var
 $product[] = array('name'=>$cart_itm["name"],
'code'=>$cart_itm["code"], 'qty'=>$cart_itm["qty"],
'price'=>$cart_itm["price"]);
 }
 }

 if($found == false) //we didn't find item in array
 {
 //add new user item in array
 $_SESSION["products"] = array_merge($product,
$new_product);
 }else{
 //found user item in array list, and increased the quantity
 $_SESSION["products"] = $product;
 }

 }else{
```

```
 //create a new session var if does not exist
 $_SESSION["products"] = $new_product;
 }

}

//redirect back to original page
header('Location:'. $return_url);
}

//remove item from shopping cart
if(isset($_GET["removep"]) && isset($_GET["return_url"]) &&
isset($_SESSION["products"]))
{
 $product_code = $_GET["removep"]; //get the product code to remove
 $return_url = base64_decode($_GET["return_url"]); //get return url

 foreach ($_SESSION["products"] as $cart_itm) //loop through session
array var
 {
 if($cart_itm["code"]!=$product_code){ //item does,t exist in the
list
 $product[] = array('name'=>$cart_itm["name"],
'code'=>$cart_itm["code"], 'qty'=>$cart_itm["qty"],
'price'=>$cart_itm["price"]);
 }

 //create a new product list for cart
 $_SESSION["products"] = $product;
 }

 //redirect back to original page
 header('Location:'. $return_url);
}
?>
```

### Bước 5 : Xem tổng quan giỏ hàng

- Bây giờ chúng ta sẽ liệt kê tất cả các sản phẩm có trong giỏ hàng, với số lượng, giá tiền, và tổng tiền sẽ phải thanh toán trước khi tiến hành ” đặt mua “. Nếu sinh viên muốn nâng cao, ở phần này có thể chèn thêm tiền thuế, phí giao hàng... Sinh viên copy đoạn code sau vào file **View\_cart.php**.

```
<?php
session_start();
include_once("config.php");
if(isset($_SESSION["products"]))
```

```
{
 $total = 0;
 echo '<form method="post" action="PAYMENT-GATEWAY">';
 echo '';
 $cart_items = 0;
 foreach ($_SESSION["products"] as $cart_itm)
 {
 $product_code = $cart_itm["code"];
 $results = $mysqli->query("SELECT product_name,product_desc,
price FROM products WHERE product_code='$product_code' LIMIT 1");
 $obj = $results->fetch_object();

 echo '<li class="cart-itm">';
 echo '<a
href="cart_update.php?removep='.$cart_itm["code"].'&return_url='.$current_u
rl.'">×';
 echo '<div class="p-price">'.$currency.$obj->price.'</div>';
 echo '<div class="product-info">';
 echo '<h3>'.$obj->product_name.' (Code :'.$product_code.'</h3>
';

 echo '<div class="p-qty">Qty : '.$cart_itm["qty"].'</div>';
 echo '<div>'.$obj->product_desc.'</div>';
 echo '</div>';
 echo '';
 $subtotal = ($cart_itm["price"]*$cart_itm["qty"]);
 $total = ($total + $subtotal);

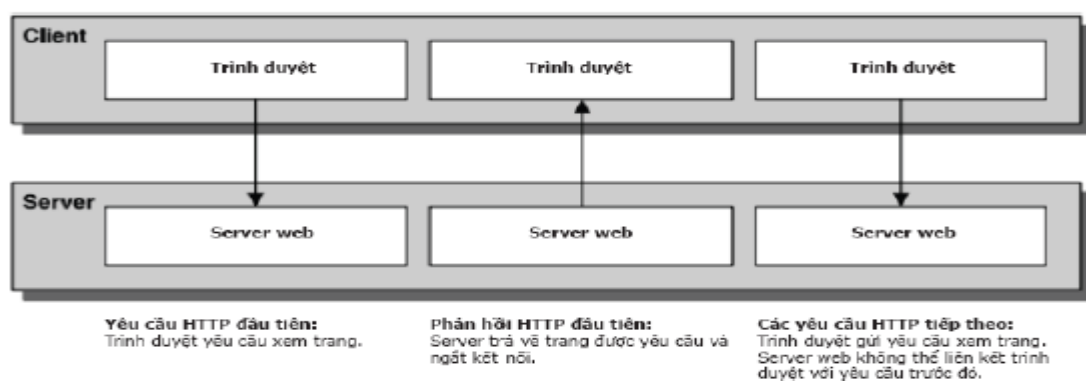
 echo '<input type="hidden" name="item_name['.$cart_items.']"
value="'.$obj->product_name.'" />';
 echo '<input type="hidden" name="item_code['.$cart_items.']"
value="'.$product_code.'" />';
 echo '<input type="hidden" name="item_desc['.$cart_items.']"
value="'.$obj->product_desc.'" />';
 echo '<input type="hidden" name="item_qty['.$cart_items.']"
value="'.$cart_itm["qty"].'" />';
 $cart_items++;
 }
 echo '';
 echo '';
 echo 'Total : '.$currency.$total.' ';
 echo '';
 echo '</form>';

}
else{
 echo 'Your Cart is empty';
}
?>
```

### 7.2 Session.

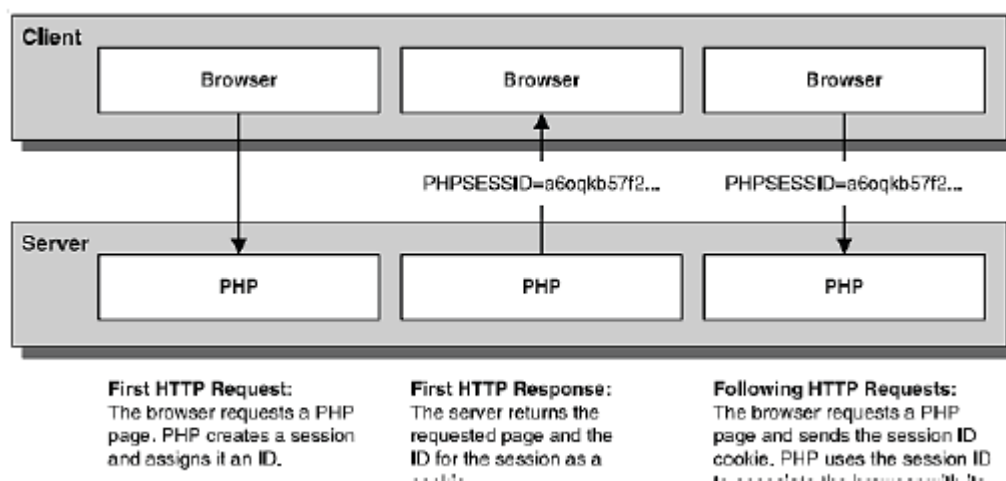
### 7.2.1 Nguyên nhân khó sử dụng Session với HTTP

- Session (phiên làm việc): sau khi trình duyệt gửi yêu cầu thì HTTP ngắt kết nối tới máy chủ. Để duy trì trạng thái, ứng dụng web phải thực hiện session.
- Mặc định, PHP sử dụng cookie để lưu session ID trên mỗi trình duyệt. Khi đó, trình duyệt chuyển cookie tới máy chủ với mỗi yêu cầu.
- Để theo dõi session khi cookie bị tắt, có thể sử dụng cách mã hóa URL để lưu session ID trong URL cho mỗi trang của ứng dụng.



### 7.2.2 Cách thức hoạt động của session trong PHP

- Bước 1: trình duyệt trên máy khách gửi yêu cầu xem trang PHP tới máy chủ Web.
- Bước 2: PHP kiểm tra xem yêu cầu đã bao gồm session ID (định danh phiên làm việc) chưa. Nếu chưa, PHP sẽ tạo ra một session mới trên máy chủ và gán cho nó một session ID duy nhất. Lúc này, ứng dụng có thể lưu dữ liệu vào session.
- Bước 3: session ID sẽ được gửi trả lại trình duyệt như cookie trong phản hồi. Khi trình duyệt gửi các yêu cầu sau đó, PHP cũng kiểm tra liệu yêu cầu đã bao gồm session ID chưa. Vì lần này session ID đã có, nên PHP sử dụng session ID này để truy cập, điều chỉnh hoặc thêm dữ liệu khi cần.



### 7.2.3 Bắt đầu một session

- Session bắt đầu khi trình duyệt gửi yêu cầu xem trang đầu tiên với hàm `session_start`

Hàm `sessionstart()`: bắt đầu phiên làm việc mới hoặc tiếp tục phiên cũ. Trả về TRUE nếu thành công và FALSE nếu ngược lại. Hàm này phải được gọi trước khi trang gửi bất kỳ dữ liệu HTML nào tới ứng dụng.

- Mặc định, session làm việc sử dụng cookie theo phiên để liên kết trình duyệt với dữ liệu. Tuy nhiên có thể sử dụng hàm `session_set_cookie_params` để tùy chỉnh cookie của phiên làm việc.

Hàm `session_set_cookie_params` phải được gọi trước hàm `session_start`.

Cú pháp hàm **`session_set_cookie_params`**:

**`session_set_cookie_params($lifetime, $path, $domain, $secure, $httponly)`**

`$lifetime`: vòng đời của cookie theo phiên làm việc tính theo giây. Mặc định là 0.

`$path`: đường dẫn trên máy chủ mà cookie theo phiên làm việc có hiệu lực. Mặc định là đường dẫn hiện tại của đoạn mã thiết lập cookie.

`$domain`: tên miền mà cookie có hiệu lực. Mặc định là tên máy chủ thiết lập cookies.

\$secure: Nếu TRUE, cookie chỉ sẵn dùng nếu được gửi qua kết nối bảo mật HTTP (kết nối HTTPS). Mặc định là FALSE.

\$httponly: Nếu TRUE, cookie chỉ sẵn dùng qua giao thức HTTP, chứ không qua các ngôn ngữ kịch bản phía khách như JavaScript. Mặc định là FALSE.

- Ví dụ: \$lifetime = 60\*60\*24\*365;// 1 năm theo giây

```
session_set_cookie_params($lifetime, '/');
```

```
sessionstart();
```

### 7.2.4 Gán và lấy giá trị của biến session

- Khi khởi tạo phiên, sử dụng biến toàn cục tự động \$\_SESSION để thiết lập và lấy dữ liệu của người dùng cho phiên làm việc. Biến này là một mảng liên kết.

- Sử dụng hàm isset để kiểm tra sự tồn tại của một phần tử trong mảng \$\_SESSION.

- Xóa nội dung mảng \$\_SESSION: thiết lập thành mảng rỗng.

+ **Hướng dẫn thiết lập và lấy biến vô hướng:**

```
// Thiết lập biến vô hướng trong phiên làm việc
```

```
$_SESSION['product_code'] = 'MBT-1753';
```

```
// Lấy biến từ phiên
```

```
$product_code = $_SESSION['product_code'];
```

+ **Hướng dẫn thiết lập và lấy mảng:**

```
// Thiết lập mảng trong phiên làm việc
```

```
if (!isset($_SESSION['cart'])) {
```

```
 $_SESSION['cart'] = array();
```

```
}
```

```
// Thêm phần tử vào mảng được lưu trong phiên làm việc
```

```
$_SESSION['cart']['key1'] = 'value1';
$_SESSION['cart']['key2'] = 'value2';

// Lấy và sử dụng mảng được lưu trong phiên

$cart = $_SESSION['cart'];

foreach ($cart as $item){ echo $item; }
```

+ **Hướng dẫn thiết lập và lấy mảng:**

```
// Xóa một biến

unset ($_SESSION['cart']);

// Xóa tất cả các biến

$_SESSION= array();
```

### 7.2.5 Kết thúc một session

- Để kết thúc một session sử dụng hàm `session_destroy()`: kết thúc phiên làm việc. Trả về TRUE nếu thành công và FALSE nếu ngược lại.
- Session kết thúc khi người dùng tắt trình duyệt, một khoảng thời gian nhất định trôi qua mà không có yêu cầu nào.
- Xóa tất cả dữ liệu liên quan đến phiên làm việc khỏi máy khách và máy chủ khi kết thúc một session

+ Xóa dữ liệu phiên làm việc khỏi bộ nhớ.

+ Gọi hàm `session_destroy`

+ Sử dụng hàm `setcookie`

- Hàm `session_name` lấy tên của cookie theo phiên làm việc. Mặc định, cookie theo phiên làm việc có tên là “PHPSESSID”.

- Hàm `session_get_cookie_params` lấy mảng liên kết chứa tất cả các tham số của cookie theo phiên

- Ví dụ: // Kết thúc session

```
$_SESSION = array();// xóa dữ liệu về session khỏi bộ nhớ.

session_destroy(); // xóa session ID

// Xóa cookie session khỏi trình duyệt

$name = session_name(); // lấy tên của cookie theo session.

$expire = strtotime('-1 year');// tạo ngày hết hạn trong quá khứ

$params = session_get_cookie_params(); // lấy tham số session

$path = $params['path'];

$domain = $params['domain'];

$secure = $params['secure'];

$httponly = $params['httponly'];

setcookie($name, '', $expire, $path, $domain, $secure, $httponly);
```

### 7.2.6 Quản lý session

- Hàm `session_name()`: lấy tên của cookie theo phiên làm việc. Mặc định là PHPSESSID.

- Hàm `session_id([$id])`: Nếu tham số không được chỉ định, hàm sẽ lấy session ID hiện hành. Nếu không có phiên làm việc nào tồn tại, hàm sẽ lấy chuỗi rỗng. Nếu tham số được chỉ định, hàm sẽ gán một giá trị cụ thể cho session ID.

- Hàm `session_write_close()`: Kết thúc phiên làm việc hiện hành và lưu dữ liệu của phiên hiện hành. Hàm này chỉ cần trong những trường hợp đặc biệt như chuyển hướng.

- Hàm `session_regenerate_id()`: Tạo session ID mới cho phiên làm việc hiện hành. Trả về TRUE nếu thành công và FALSE nếu ngược lại. Hàm này có thể được sử dụng để tránh hiện tượng chiếm đoạt phiên.

- Nếu chuyển hướng trình duyệt bằng cách dùng hàm header theo sau là hàm `exit`, không phải lúc nào PHP cũng lưu dữ liệu về session. Vì vậy, cần gọi hàm `session_write_close` để buộc PHP lưu dữ liệu về session.

**- Ví dụ:**

```
// Lấy tên cookie theo phiên làm việc
$name = session_name(); // mặc định là PHPSESSID

// Lấy giá trị của session ID
$id = session_id(); // ví dụ: 11je1flfo1g8und

// Thiết lập session ID
session_id('abc123');
```

### 7.3 Cookie.

#### 7.3.1 Giới thiệu Cookie

- Cookie là cặp tên/giá trị được lưu trên trình duyệt máy khách

Ví dụ một cookie:

```
PHPSESSID=D1F15245171203E8670487F020544490
user_id=87
email=j_smi_thehotmail.com userName=jsmith
passwordCookie=opensesame
```

- Những cách dùng thường gặp của cookie:

+ Cho phép người dùng không cần đăng nhập và điền form đăng ký. Thu thập những dữ liệu như tên người dùng, mật khẩu, địa chỉ hoặc thông tin thẻ tín dụng.

+ Tùy chỉnh trang hiển thị các thông tin như dự báo thời tiết, tỷ số các trận đấu thể thao và giá cổ phiếu.

- + Quảng cáo trúng đích: Các banner quảng cáo nhắm vào khách hàng mục tiêu quan tâm đến quảng cáo
- Trên máy chủ, ứng dụng Web tạo cookie và gửi nó tới trình duyệt. Trên máy khách, trình duyệt lưu cookie và gửi nó trở lại máy chủ mỗi khi truy cập trang từ máy chủ đó. Mặc định, cookie chỉ có hiệu lực cho đến khi người dùng đóng trình duyệt. Tuy nhiên, có thể thiết lập để cookie tồn tại trong trình duyệt của người dùng với thời gian lên đến 3 năm. Mỗi cookie được giới hạn 4KB dung lượng.

### 7.3.2 Thiết lập và truy xuất cookie

- Sử dụng hàm setcookie với cú pháp:

Setcookie(\$name, \$value, \$expire, \$path, \$domain, \$secure, \$httponly)

\$name: tên cookie

\$value: Giá trị của cookie. Mặc định là chuỗi rỗng.

\$expire: ngày hết hạn của cookie là nhãn thời gian. Nếu được thiết lập là 0, cookie sẽ hết hạn khi người dùng tắt trình duyệt, Giá trị mặc định là 0.

\$path: Đường dẫn trên máy chủ mà cookie có hiệu lực. Nếu được thiết lập là /, cookie sẽ có hiệu lực với tất cả các đường dẫn trên máy chủ hiện tại. Đường dẫn mặc định là đường dẫn cho file thiết lập cookie.

\$domain: Tên miền mà cookie có hiệu lực. Mặc định là tên của máy chủ thiết lập cookie.

\$secure: Nếu là TRUE, cookie sẽ chỉ có hiệu lực nếu được gửi qua HTTPS. Mặc định là FALSE.

\$httponly: Nếu là TRUE, cookie chỉ có hiệu lực qua giao thức HTTP chứ không qua các ngôn ngữ kịch bản phía máy khách như JavaScript. Mặc định là FALSE.

- Ví dụ:

```
// Thiết lập cookie trong trình duyệt
$name = 'userid';
$value = 'rharris';
$expire = strtotime('+ 1 year');
```

```
$path = ' / ' ;
setcookie($name, $value, $expire, $path);
// Lấy giá trị của cookie từ trình duyệt
$userid = $COOKIE['userid']; // $userid là 'rharris'
// Xóa cookie khỏi máy chủ
$expire = strtotime('-1 year');
setcookie('userid', ' ', $expire, ' / ');
```

### 7.3.3 Bật và tắt cookie

- Để kiểm thử hoạt động của ứng dụng trong trường hợp người dùng tắt cookie, bạn có thể tắt cookie trên trình duyệt của mình.

- Để kiểm thử ứng dụng trong điều kiện bình thường, bật cookie trên trình duyệt.
- Hướng dẫn bật tắt cookie trong trình duyệt Firefox
  - Mở menu Tools và chọn lệnh Options.
  - Nhấn tab Privacy.
  - Sử dụng check box “Accept cookies from sites” để bật hoặc tắt cookie.
- Hướng dẫn bật tắt cookie trong trình duyệt Internet Explorer
  - Mở menu Tools và chọn lệnh Options.
  - Nhấn tab Privacy.
  - Sử dụng thanh trượt điều khiển để bật hoặc tắt cookie. Để tắt cookie, thiết lập cấp bảo mật là “Block All Cookies”. Để bật cookie, nhấn vào nút Default nhằm quay về thiết lập bảo mật mặc định.

### Tổng kết bài học

- Cookie là một cặp tên/giá trị được lưu trong trình duyệt. Cookie theo phiên làm việc sẽ hết hạn khi người dùng đóng trình duyệt.
- Một số trình duyệt không hỗ trợ cookie.
- Vì HTTP là giao thức phi trạng thái, nên hầu hết ứng dụng Web cần session để kiểm soát phiên làm việc của mỗi người dùng.

- Mặc định, PHP thực hiện theo dõi phiên làm việc bằng việc lưu cookie trên trình duyệt của mỗi người với một session ID duy nhất. Sau đó session ID này có thể được dùng để truy cập dữ liệu dành cho phiên làm việc đó.

Để thiết lập hoặc truy xuất dữ liệu cho một phiên làm việc thì sử dụng biến toàn cục tự động \$\_SESSION. Biến này là một mảng liên kết.

- PHP cung cấp các hàm để khởi tạo, kết thúc và quản lý session.

### B.CÁC DẠNG BÀI TẬP

**BT6-1.** Sinh viên sử dụng công cụ Cpanel để đưa Website lên host.

**Hướng dẫn giải bài tập 6-1:** Sinh viên thực hiện theo các bước sau đây:

**Bước 1:** Truy cập vào Cpanel

- Có thể truy cập vào Cpanel theo một trong hai địa chỉ sau:

<http://www.yourdomain.com/config>

<http://www.yourdomain.com:2222>

**Bước 2:** Upload Website lên server

- Các thư mục trên host bao gồm:

/backups

/domains

/mail

/public\_html

- Trang mặc định (index.html/index.php)

File mặc định được chạy cho thư mục Web là file index.html. Nghĩa là khi Website được truy cập bởi tên miền <http://www.yourdomain.com>, server sẽ trả về trang <http://www.yourdomain.com/index.php> hoặc index.html. Điều này được áp dụng cho tất cả các thư mục có thể truy nhập được thông qua domain bao gồm cả subdomain.

- Sử dụng FTP để Upload files

- Cài đặt phần mềm FTP client để Upload file lên server. Sinh viên có thể download một số phần mềm sau: CuteFTP, WS\_FTPLE, FTP Explorer, ...

- Kết nối đến Server

FTP clients cần sử dụng các thông tin sau:

Username: Đây là username Cpanel của người dùng.

Password: Đây là password Cpanel của người dùng.

Address: Đây là địa chỉ của Server.

Port: Sử dụng port mặc định (21).

- Mở thư mục public\_html sinh viên upload các file và thư mục của Website vào đây.

**Bước 3:** Mở trình duyệt gõ địa chỉ trang Web và theo dõi Website của mình.

**BT7-1.** Xây dựng form xây dựng giỏ hàng Shopping Cart cho phép người sử dụng mua hàng qua mạng bao gồm các bước như mục **7.1 Xây dựng giỏ hàng**

**Hướng dẫn giải bài tập 7-1:** Sinh viên thiết kế form giao diện và viết chương trình theo các bước sau đây:

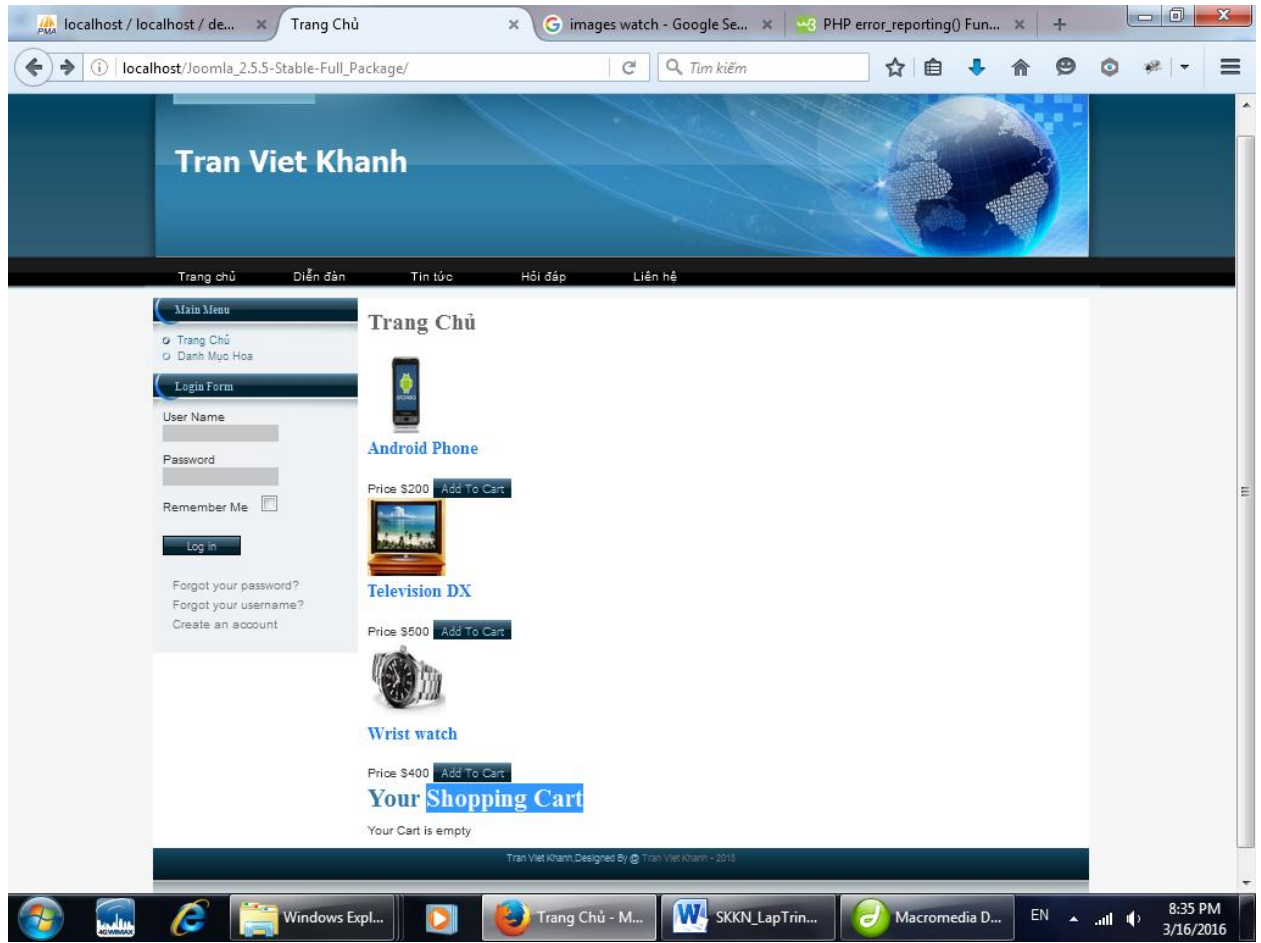
```
<?php
 error_reporting(E_ERROR | E_WARNING | E_ALL);
 //session_start();
 include_once("Config.php");
?>
<div class="products">
 <?php
 //current URL of the Page. cart_update.php redirects back to this URL
 $current_url =
 base64_encode("http://".$_SERVER['HTTP_HOST'].$_SERVER['REQUEST_URI'])
 ;

 $results = $mysqli->query("SELECT * FROM products ORDER BY id ASC");
 if ($results) {
 //output results from database
 while($obj = $results->fetch_object())
 {

 echo '<div class="product">';
 echo '<form method="post" action="cart_update.php">';
 echo '<div class="product-thumb"></div>';
 echo '<div class="product-content"><h3>'.$obj->product_name.'</h3>';
 echo '<div class="product-desc">'.$obj->product_desc.'</div>';
 echo '<div class="product-info">Price '.$currency.$obj->price.' <button
class="add_to_cart">Add To Cart</button></div>';
 echo '</div>';
 echo '<input type="hidden" name="product_code" value="'.$obj->
product_code.'" />';
 echo '<input type="hidden" name="type" value="add" />';
 echo '<input type="hidden" name="return_url" value="'.$current_url.'" />';
 echo '</form>';
 echo '</div>';
 }
 }
 }
```

```
?>
</div>
<div class="shopping-cart">
 <h2>Your Shopping Cart</h2>
 <?php
 if(isset($_SESSION["products"]))
 {
 $total = 0;
 echo '';
 foreach ($_SESSION["products"] as $cart_itm)
 {
 echo '<li class="cart-itm">';
 echo '&t
imes;';
 echo '<h3>'.$cart_itm["name"].'</h3>';
 echo '<div class="p-code">P code : '.$cart_itm["code"].'</div>';
 echo '<div class="p-qty">Qty : '.$cart_itm["qty"].'</div>';
 echo '<div class="p-price">Price : '.$currency.$cart_itm["price"].'</div>';
 echo '';
 $subtotal = ($cart_itm["price"]*$cart_itm["qty"]);
 $total = ($total + $subtotal);
 }
 echo '';
 echo 'Total : '.$currency.$total.' Check-out!';
 echo 'Empty
Cart';
 }else{
 echo 'Your Cart is empty';
 }
 ?>
</div>
```

- **Kết quả khi chạy chương trình sẽ giống như hình sau:**



## C.BÀI TẬP NÂNG CAO

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

**BT7-2.** Xây dựng giỏ hàng (shopping cart) như bài tập 7-1 nhưng có thêm chức năng cho phép cập nhật giỏ hàng (update cart) và hiển thị giỏ hàng (view cart)

**Gợi ý làm bài:** Thiết kế form giao diện như **BT7-1**:

- Viết code cập nhật giỏ hàng như đã nêu ở phần lý thuyết Bước 4: Cập nhật giỏ hàng (Mục 7.1).
- Viết code hiển thị giỏ hàng như đã nêu ở phần lý thuyết Bước 5: Hiển thị giỏ hàng (Mục 7.1).

## KẾT LUẬN

### **a/. Ý nghĩa khoa học:**

- Đề tài đã trình bày tóm tắt cơ sở lý thuyết và xây dựng hệ thống bài tập tự học thực hành giúp cho sinh viên tư duy sáng tạo trong lập trình ngôn ngữ PHP, hoàn thiện dần các kỹ năng lập trình, phong cách viết chương trình, tiến tới xây dựng các ứng dụng web một cách hoàn chỉnh.
- Đề tài giúp cho sinh viên có một tài liệu để học tập tốt và là tài liệu hỗ trợ cho sinh viên tự học có kết quả cao.

### **b/. Ý nghĩa thực tiễn:**

- Giúp cho giảng viên thuận tiện trong việc giảng dạy lý thuyết và giảng dạy thực hành, thuận tiện trong việc đánh giá kết quả học tập của sinh viên.
- Giúp cho giảng viên hoàn thiện hơn về kỹ năng lập trình ngôn ngữ PHP, khả năng trình bày, giúp cho việc trao đổi chuyên môn của các giảng viên dễ dàng hơn. Từ đó có thể được nhân rộng ra để biên soạn các đề tài khác nhằm mục đích nâng cao chất lượng đào tạo sinh viên của trường.

### **c/. Mục đích nghiên cứu:**

- Hoàn thiện kiến thức kỹ năng về lập trình PHP, nâng cao khả năng trình bày, khả năng sư phạm, thuận tiện trong việc giảng dạy lý thuyết, thực hành, và cả trong khâu đánh giá sinh viên.
- Cung cấp tài liệu để sinh viên tự học có kết quả cao, có cơ sở để học tốt các học phần tiếp theo.
- Giúp sinh viên phát triển tư duy lập trình, khả năng sáng tạo, rèn luyện kỹ năng và phong cách viết chương trình.
- Giúp cho khoa nâng cao công tác nghiên cứu khoa học cơ bản của các giảng viên, thuận tiện trong việc trao đổi, thống nhất về chuyên môn trong giảng dạy.
- Bổ sung tài liệu giảng dạy cho kho giáo trình của nhà trường, làm tài liệu tham khảo cho các giảng viên dạy môn lập trình PHP.