**Bài tập thực hành 02**

# Làm quen với ngôn ngữ PHP

Họ tên: Ngô Như Huỳnh

MSSV: B2110044

**Mục tiêu cần đạt**:

Sau buổi các bạn cần nắm thao tác cơ bản với PHP:

* Các thành phần trong PHP, các cấu trúc lập trình cơ bản
* Tạo 1 form cơ bản với PHP
* Khai báo và khởi tạo class

Các bạn đọc và thực hiện viết code như hướng dẫn để làm quen, sau đó đọc trả lời các **Yêu cầu** ở dưới bài hướng dẫn để thực hiện yêu cầu bài thực hành.

**Gợi ý Tham khảo:**

* <https://www.w3schools.com/php/>
* và các nguồn khác mà sinh viên tìm được

## Giới thiệu

PHP là một ngôn ngữ lập trình, như trong [[1]](#footnote-1), PHP được sử dụng rất rộng rãi, và các script PHP chạy trên các web server (cần cài web server để thực thi), dễ download và sử dụng. Khi cài xong xampp, chúng ta đã có sẵn web server là Apache để thực thi script PHP.

### Tạo file php đầu tiên

Script php có thể đặt bất cứ đâu trong file (với đuôi là .php)

Code được bao bởi <?php ?>

<?php

// PHP code goes here

?>

Vd: tạo tập tin hello.php hiển thị chữ “Hello!”

<!DOCTYPE html>

<html>

<body>

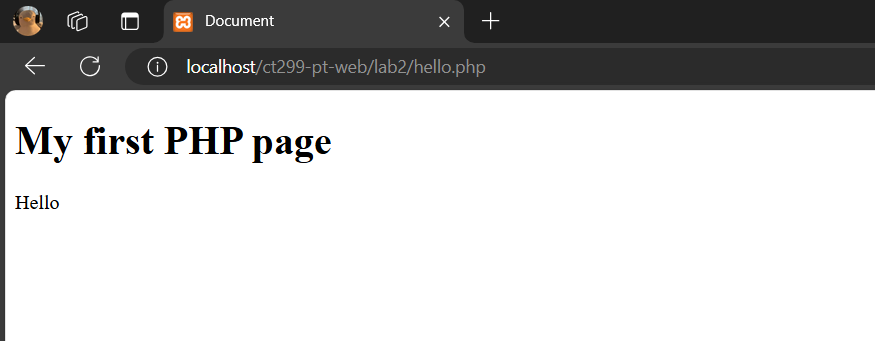
<h1>My first PHP page</h1>

<?php echo "Hello!";

?>

</body>

</html>



### Cấu trúc lập trình trong php

#### If

Ví dụ, bên dưới chúng ta tạo cấu trúc if, elseif, và else xét biết t có kiểu giờ hiện hành, và xét theo giờ hiện để hiện thị thông báo

Tạo if.php có nội dung sau:

<?php

date\_default\_timezone\_set('Asia/Ho\_Chi\_Minh');

//https://www.php.net/manual/en/timezones.asia.php

$t = date("H");

echo 'Bay gio la: '.$t.' gio';

$s=''; if ($t < 11) {

$s="sang";

} elseif ($t < 17) {

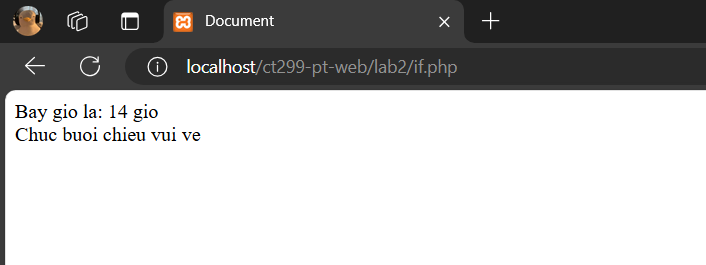
$s= "chieu";

} else {

$s ="toi"; }

echo '<br> Chuc buoi '. $s. ' vui ve';

?>



#### Switch

Ví dụ, bên dưới chúng ta tạo cấu trúc switch xét biết favcolor có chứa giá trị gì để hiển thị thông báo

Tạo switch.php có nội dung sau:

<?php

$favcolor = "red";

switch ($favcolor) { case "red": echo "Your favorite color is red!"; break; case "blue":

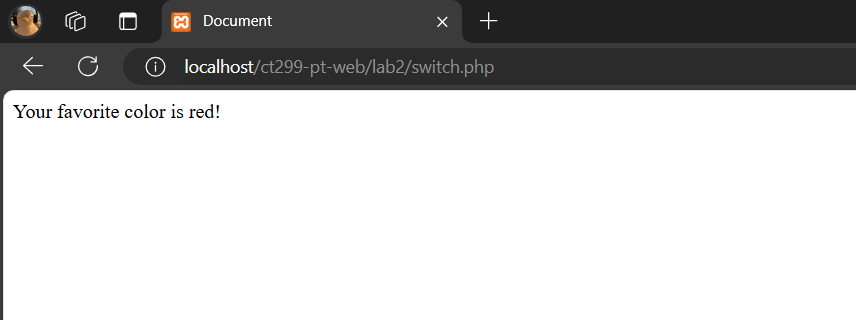
echo "Your favorite color is blue!"; break; case "green":

echo "Your favorite color is green!"; break; default:

echo "Your favorite color is neither red, blue, nor green!";

}

?>



#### Loop

PHP hỗ trợ nhiều dạng loop như while, for, do … while, foreach

Vd: tạo file loop.php viết các lệnh sau và quan sát kết quả

<?php

$x = 1; echo "while: <br>"; while($x <= 5) { echo "The number is: $x <br>";

$x++;

}

echo "do… while: <br>";

$x = 1;

do { echo "The number is: $x <br>";

$x++;

} while ($x <= 5);

echo "for: <br>";

$x = 1; for ($x = 0; $x <= 10; $x++) { echo "The number is: $x <br>";

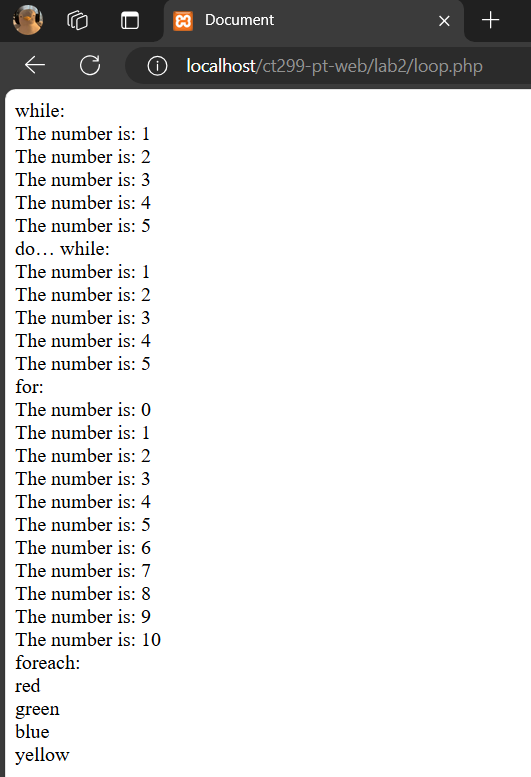
} echo "foreach: <br>";

$colors = array("red", "green", "blue", "yellow");

foreach ($colors as $value) { echo "$value <br>";

}

?>



### Định nghĩa hàm

Tạo tập tin function.php với lệnh sau để trả về kết quả cộng 2 biến

<?php function addNumbers(int $a, int $b) { return $a + $b;

}

echo addNumbers(5, 10); // it will return 15 echo addNumbers(5, “so 10”);

// neu 1 tham so la chuoi co the se xuat thong bao loi

?>

Nếu bị lỗi do tham số có kiểu không tương thích ta thêm declare(strict\_types=0); lúc này không thông báo lỗi (áp dụng cho PHP 7)

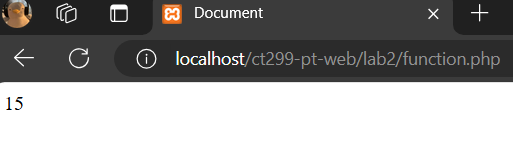
<?php declare(strict\_types=0); function addNumbers(int $a, int $b) { return $a + $b;

}

echo addNumbers(5, 10); // it will return 15 echo addNumbers(5, “so 10”);

// neu 1 tham so la chuoi co the se xuat thong bao loi

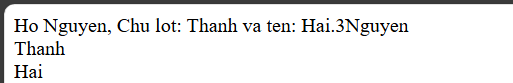
?>



### Mảng

Để lưu trữ nhiều giá trị trong 1 biến, ví dụ tạo tập tin array.php với các lệnh sau:

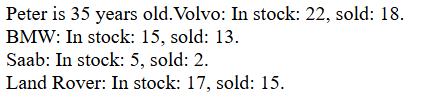
* **Sử dụng mảng và in giá trị**

****

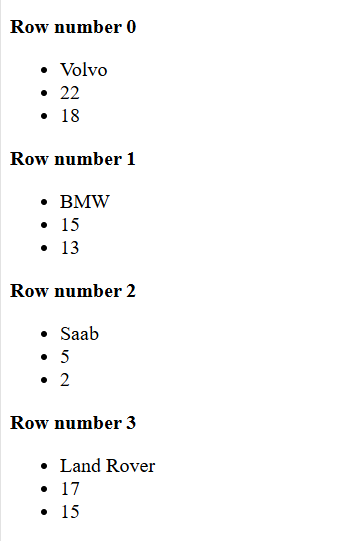
* **Sử dụng mảng kiểu “gom key => giá trị”**

****

* **Sử dụng mảng đa chiều**

****

* **Hiển thị mảng đa chiều qua for**

****

### Xử lý với form

Tạo tập tin welcome.php

<html>

<body>

Hello <?php echo $\_POST["name"]; ?><br>

Your email address is: <?php echo $\_POST["email"]; ?>

</body>

</html>

Tạo tập tin nhap.php

<!DOCTYPE HTML>

<html>

<body>

<form action="welcome.php" method="post">

Name: <input type="text" name="name"><br>

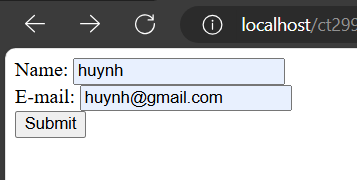
E-mail: <input type="text" name="email"><br>

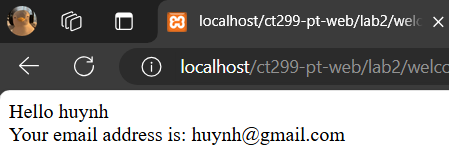
<input type="submit">

</form>

</body>

</html>





Lưu 2 tập tin trên vào thư mục nào đó trong xampp/htdocs, vd: php1 Chạy tập tin gõ trên thanh trình duyệt localhost/php1/nhap.php

K

ế

t qu

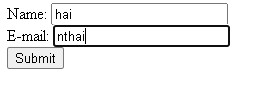
ả

hi

ể

n th

ị



### Kiểm tra dữ liệu trên form

Tạo tập tin sau: validation.php

<!DOCTYPE HTML>

<html>

<head>

</head>

<body>

<?php

// define variables and set to empty values

$name = $email = $gender = $comment = $website = "";

if ($\_SERVER["REQUEST\_METHOD"] == "POST") { $name = test\_input($\_POST["name"]);

$email = test\_input($\_POST["email"]);

$website = test\_input($\_POST["website"]); $comment = test\_input($\_POST["comment"]); if (isset($\_POST["gender"]))

$gender = test\_input($\_POST["gender"]);

} function test\_input($data) { $data = trim($data);

$data = stripslashes($data); $data = htmlspecialchars($data); return $data;

}

?>

<h2>PHP Form Validation Example</h2> <form method="post" action="<?php echo htmlspecialchars($\_SERVER["PHP\_SELF"]);?>">

Name: <input type="text" name="name">

<br><br>

E-mail: <input type="text" name="email">

<br><br>

Website: <input type="text" name="website">

<br><br>

Comment: <textarea name="comment" rows="5" cols="40"></textarea> <br><br> Gender:

<input type="radio" name="gender" value="female">Female

<input type="radio" name="gender" value="male">Male

<input type="radio" name="gender" value="other">Other

<br><br>

<input type="submit" name="submit" value="Submit">

</form>

<?php echo "<h2>Your Input:</h2>"; echo $name; echo "<br>"; echo $email; echo "<br>"; echo $website; echo "<br>"; echo $comment; echo "<br>"; echo $gender;

?>

</body>

</html>

Yêu cầu nhập liệu trên form

Tạo tập tin required.php

<!DOCTYPE HTML>

<html>

<head>

<style>

.error {color: #FF0000;}

</style>

</head>

<body>

<?php

// define variables and set to empty values

$nameErr = $emailErr = $genderErr = $websiteErr = "";

$name = $email = $gender = $comment = $website = "";

if ($\_SERVER["REQUEST\_METHOD"] == "POST") { if (empty($\_POST["name"])) {

$nameErr = "Name is required";

} else {

$name = test\_input($\_POST["name"]);

} if (empty($\_POST["email"])) {

$emailErr = "Email is required";

} else {

$email = test\_input($\_POST["email"]);

} if (empty($\_POST["website"])) {

$website = "";

} else {

$website = test\_input($\_POST["website"]);

} if (empty($\_POST["comment"])) {

$comment = "";

} else {

$comment = test\_input($\_POST["comment"]);

}

if (empty($\_POST["gender"])) {

$genderErr = "Gender is required";

} else {

$gender = test\_input($\_POST["gender"]);

}

}

function test\_input($data) { $data = trim($data);

$data = stripslashes($data);

$data = htmlspecialchars($data);

return $data;

}

?>

<h2>PHP Form Validation Example</h2>

<p><span class="error">\* required field</span></p>

<form method="post" action="<?php echo htmlspecialchars($\_SERVER["PHP\_SELF"]);?>">

Name: <input type="text" name="name">

<span class="error">\* <?php echo $nameErr;?></span>

<br><br>

E-mail: <input type="text" name="email">

<span class="error">\* <?php echo $emailErr;?></span>

<br><br>

Website: <input type="text" name="website">

<span class="error"><?php echo $websiteErr;?></span>

<br><br>

Comment: <textarea name="comment" rows="5" cols="40"></textarea> <br><br> Gender:

<input type="radio" name="gender" value="female">Female

<input type="radio" name="gender" value="male">Male

<input type="radio" name="gender" value="other">Other

<span class="error">\* <?php echo $genderErr;?></span>

<br><br>

<input type="submit" name="submit" value="Submit">

</form>

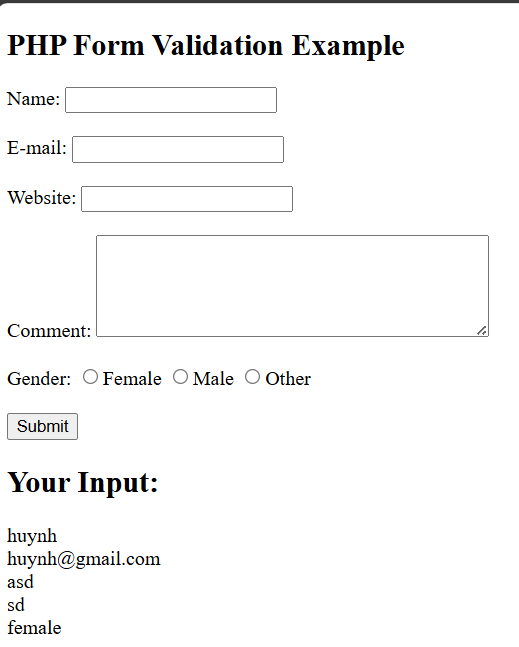
<?php echo "<h2>Your Input:</h2>"; echo $name; echo "<br>"; echo $email; echo "<br>"; echo $website; echo "<br>"; echo $comment; echo "<br>"; echo $gender;

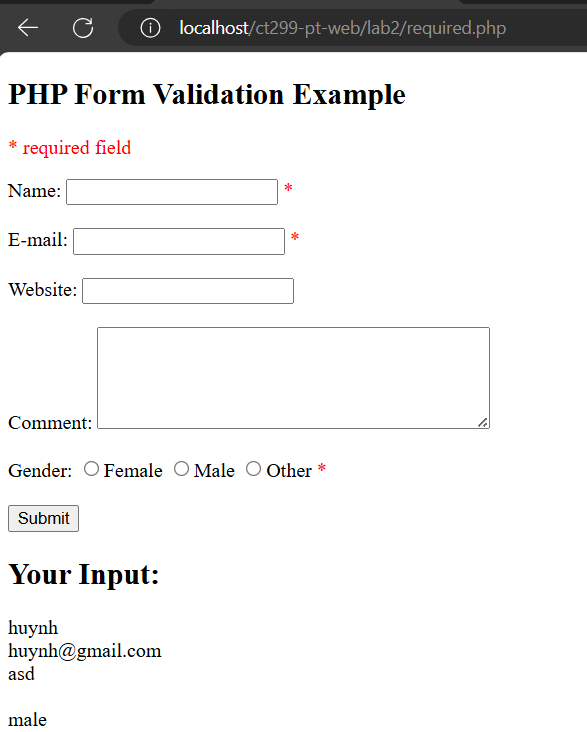
?>

</body>

</html>

Khi chúng ta không nhập gì mà bấm submit sẽ báo lỗi





### Hướng đối tượng (OOP) trong PHP

Vd: tạo 1 tập tin fruit\_class.php

<?php

//khai bao 1 lop fruit class Fruit { //gom 2 thuoc tinh public $name; public $color;

// Methods, phuong thuc trong lop //gan gia tri cho thuoc tinh function set\_name($name) {

$this->name = $name;

} //tra ve gia tri function get\_name() { return $this->name; }

//phuong thuc xay dung function \_\_construct($name) {

$this->name = $name;

}

//phuong thuc huy

function \_\_destruct() { echo "The fruit is {$this->name}.";

}

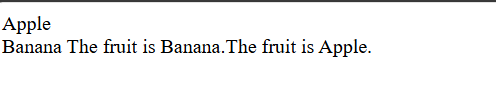
}

// khoi tao bien kieu class Fruit() voi ham xay dung

$apple = new Fruit(‘Apple’);

$banana = new Fruit(‘Banana’);

//hien thi cac gia tri thong qua cac phuong thuc da dinh nghia echo $apple->get\_name(); echo "<br>"; echo $banana->get\_name(); ?>



Yêu cầu bài thực hành:

1. Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.
2. Hãy cho biết cách khai báo biến, và gán trị cho biến trong PHP. Liệt kê các chú ý khi đặt

tên biến. Cho các ví dụ về khai báo biến, và gán giá trị kiểu chuỗi, kiểu số, ngày tháng.

* **Khai báo biến:**

+ Sử dụng dấu đô-la ($) để bắt đầu đặt tên biến.

+ Tên biến phải bắt đầu bằng một chữ cái hoặc dấu gạch dưới (\_).

+ Tên biến có thể chứa các chữ cái, số và dấu gạch dưới.

+ Tên biến phải phân biệt chữ hoa và chữ thường.

**+ Ví dụ: $caTim; $so4; $phao**

* **Gán trị cho biến:** Sử dụng toán tử gán “=” để gán giá trị cho biến.

+ Ví dụ $a = 10; $chuoi=”hello”

* **Chú ý khi đặt tên biến:**

+ Tránh sử dụng các từ khóa dành riêng của PHP (ví dụ: if, else, while, etc.).

* Sử dụng tên biến mô tả và có ý nghĩa với giá trị mà nó đang lưu trữ.
* Tuân theo quy tắc chữ hoa và chữ thường để tạo tên biến dễ đọc và hiểu.
* **Gán giá trị cho biến:**
* **Kiểu chuỗi:** *$malop = "DI789";*
* **Kiểu số:** *$soLuong = 20;*
* **Kiểu ngày, tháng:***$ngayBan = "2024-09-5";*

1. Liệt kê ký hiệu để biết bắt đầu và kết thúc khối lệnh PHP.

* **Ký hiệu bắt đầu khối lệnh PHP: <?php**
* **Ký hiệu kết thúc khối lệnh PHP: ?>**
* **Ví dụ:**

**<?php**

// Bắt đầu khối lệnh PHP

echo “Hello”;

// Kết thúc khối lệnh PHP

**?>**

1. Liệt kê các ký hiệu có thể được dùng để tạo comment (ghi chú) trong khối lệnh PHP.

* **Các ký hiệu dùng tạo comment:**
* **Comment 1 dòng: //**
* **Comment nhiều dòng: /\* khối lệnh \*/**
* **Ví dụ:**

**- Comment 1 dòng: //**

**<?php**

**// Đây là một comment trên một dòng**

Echo “Hello world”;

**?>**

* **Comment nhiều dòng: /\* khối lệnh \*/**

**<?php**

**/\* Đây là một comment**

**trên nhiều dòng \*/**

**echo “Hello world”;**

**?>**

1. Cho biết có bao nhiêu loại khối lệnh để thực hiện vòng lặp. Cho biết sự khác biệt của foreach với các loại khối lệnh vòng lặp khác. Cho biết cách dùng của lệnh “continue” và lệnh “break” trong các vòng lặp.

* **Các loại khối lệnh trong vòng lặp:** có 5 loại **[**for, while, do-while, foreach, foreach với key và value**]**
* **for:** Dùng để thực hiện một vòng lặp với điều kiện kiểm tra và sửa đổi biến đếm.
  + **Ví dụ:**

**for ($i = 0; $i < 4; $i++) {**

**// Thực hiện các công việc trong vòng lặp**

**echo $i;**

**}**

* **while:** Thực hiện một vòng lặp với điều kiện kiểm tra trước khi thực hiện mỗi lần lặp.
  + **Ví dụ:**

**$i = 0;**

**while ($i < 4) {**

**// Thực hiện các công việc trong vòng lặp**

**echo $i;**

**$i++;**

**}**

* **do-while:** Tương tự như while, nhưng kiểm tra điều kiện sau mỗi lần lặp.
  + **Ví dụ:**

**$i = 0;**

**do {**

**// Thực hiện các công việc trong vòng lặp**

**echo $i;**

**$i++;**

**} while ($i < 4);**

* **foreach:** Dùng để lặp qua mỗi phần tử của một mảng hoặc đối tượng.
  + **Ví dụ:**

**$array = [1, 2, 3, 4, 5];**

**foreach ($array as $value) {**

**// Thực hiện các công việc trong vòng lặp với mỗi phần tử**

**echo $value;**

**}**

* **foreach với key và value:** Cách sử dụng foreach để lặp qua mảng với cả key và value.
  + **Ví dụ:**

**$array = ['a' => 1, 'b' => 2, 'c' => 3];**

**foreach ($array as $key => $value) {**

**// Thực hiện các công việc trong vòng lặp với mỗi phần tử và key**

**echo "$key: $value";**

**}**

* **Sự khác biệt của foreach:** foreach được thiết kế đặc biệt để lặp qua mảng hoặc đối tượng một cách dễ dàng và tự động, giảm sự cần thiết của biến đếm và điều kiện lặp.
* **Cách dùng các lệnh:**
* **continue:**  Lệnh này dùng để bỏ qua phần còn lại của vòng lặp và chuyển đến lần lặp tiếp theo.
  + **Ví dụ:**

**for ($i = 0; $i < 5; $i++) {**

**if ($i == 2) {**

**continue; // Bỏ qua lần lặp khi $i bằng 2**

**}**

**echo $i;**

**}**

* **break:** Lệnh này dùng để kết thúc vòng lặp ngay lập tức khi được gọi.
  + **Ví dụ:**

**for ($i = 0; $i < 5; $i++) {**

**if ($i == 3) {**

**break; // Kết thúc vòng lặp khi $i bằng 3**

**}**

**echo $i;**

**}**

1. Viết 1 tập tin php tên giaithua.php, định nghĩa hàm giai thừa và chạy thử với 10!

**<?php**

**function tinhGiaiThua($n) {**

**if ($n == 0 || $n == 1)**

**return 1;**

**else**

**return $n \* tinhGiaiThua($n - 1);**

**}**

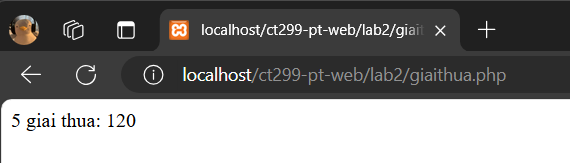
**// Chạy thử với giá trị 5!**

**$number = 5;**

**$giaithua = tinhGiaiThua($number);**

**echo "$number giai thua: $giaithua";**

**?>**

****

1. Viết 1 tập tin cong2mang.php viết hàm xử lý để cộng được 2 mảng có cùng độ dài, nếu 2 mảng không cùng độ dài thì thông báo lỗi. Chạy thử hàm với 2 mảng sau:

a={344,224,223,7737,9922,-828}, b={-344,-324,123,773,-9922,828} và cho biết kết quả hiển thị.

**<?php**

**function congHaiMang($a, $b) {**

**// Kiểm tra độ dài của hai mảng**

**$lenA = count($a);**

**$lenB = count($b);**

**if ($lenA != $lenB)**

**// Nếu độ dài không bằng nhau, thông báo lỗi**

**return "Lỗi: Hai mảng không có cùng độ dài.";**

**else {**

**// Nếu độ dài bằng nhau, thực hiện phép cộng**

**$ketQua = array();**

**for ($i = 0; $i < $lenA; $i++)**

**$ketQua[] = $a[$i] + $b[$i];**

**return $ketQua;**

**}**

**}**

**// Mảng đầu vào**

**$a = array(344, 224, 223, 7737, 9922, -828);**

**$b = array(-344, -324, 123, 773, -9922, 828);**

**// Gọi hàm và hiển thị kết quả**

**$result = congHaiMang($a, $b);**

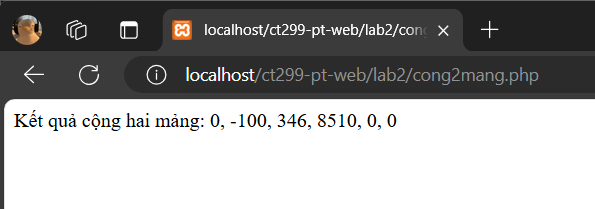
**if (is\_array($result))**

**echo "Kết quả cộng hai mảng: " . implode(', ', $result);**

**else**

**echo $result;**

**?>**

****

1. Chạy ví dụ tập tin validation.php, dựa vào link [[2]](#footnote-2), bạn hãy giải thích những bước diễn ra quá trình này.

**1. Khởi tạo biến :**

* Biến **$name, $email, $gender, $comment**, và **$website** được khởi tạo để lưu trữ dữ liệu nhập từ biểu mẫu.
* Hàm **test\_input** được định nghĩa để xử lý dữ liệu đầu vào. Nó sẽ thực hiện các bước sau đây:
* **trim**: Loại bỏ khoảng trắng ở đầu và cuối chuỗi.
* **stripslashes**: Loại bỏ các ký tự backslashes () từ chuỗi.
* **htmlspecialchars**: Chuyển đổi các ký tự đặc biệt thành các thực thể HTML.

**2. Kiểm tra và xử lý dữ liệu sau khi gửi biểu mẫu:**

* Sử dụng **$\_SERVER["REQUEST\_METHOD"]** Điều kiện này kiểm tra xem form có được gửi bằng phương thức POST hay không. Nếu có, mã bên trong khối điều kiện sẽ được thực hiện.
* Nếu biểu mẫu đã được gửi đi, tiến hành lấy dữ liệu từ các trường của biểu mẫu và thực hiện hàm **test\_input** để xử lý dữ liệu.

**3. Nhận dữ liệu từ biểu mẫu:**

* Dữ liệu được lấy từ biểu mẫu thông qua mảng $\_POST. Hàm test\_input() được gọi để xử lý các dữ liệu này, giúp loại bỏ các ký tự không mong muốn và bảo vệ ứng dụng khỏi các cuộc tấn công XSS

**4. Xây dựng biểu mẫu HTML:**

* Biểu mẫu được cấu hình để gửi dữ liệu bằng phương thức POST.
* action sử dụng htmlspecialchars($\_SERVER["PHP\_SELF"]) để đảm bảo rằng trang sẽ gửi dữ liệu đến chính nó một cách an toàn.

**5. Hiển Thị Kết Quả:**

- Sau khi dữ liệu được xử lý, kết quả sẽ được hiển thị cho người dùng. Các biến được in ra để cho người dùng thấy thông tin mà họ đã nhập.

1. Hãy sửa lại tập tin nhap.php, thêm 1 ô để nhập mật khẩu (các ký tự nhập vào sẽ biến thành dấu \*), và 1 ô để nhập ngày tháng năm sinh của bạn. Sau khi nhập xong hiển thị tên, email và mật khẩu, ngày tháng năm sinh đã nhập. Chụp màn hình sau khi bạn đã nhập dữ liệu của chính bạn, chụp màn hình hiển thị dữ liệu nhập.

* **Nhap2.php**

<!DOCTYPE HTML>

<html>

<body>

<form action="welcomev2.php" method="post">

Name: <input type="text" name="name"><br>

E-mail: <input type="text" name="email"><br>

Password: <input type="password" name="password"><br>

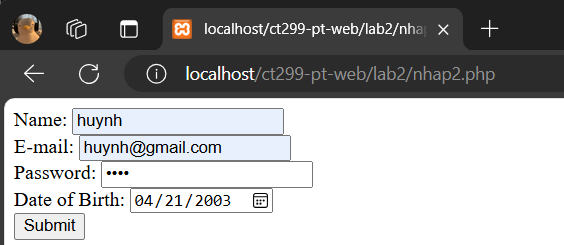
Date of Birth: <input type="date" name="dob"><br>

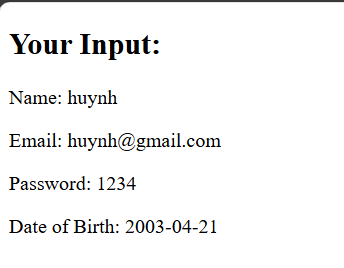
<input type="submit">

</form>

</body>

</html>





* Welcome2.php

<!DOCTYPE HTML>

<html>

<body>

<?php

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$name = test\_input($\_POST["name"]);

$email = test\_input($\_POST["email"]);

$password = test\_input($\_POST["password"]);

$dob = test\_input($\_POST["dob"]);

}

function test\_input($data) {

$data = trim($data);

$data = stripslashes($data);

$data = htmlspecialchars($data);

return $data;

}

?>

<h2>Your Input:</h2>

<p>Name: <?php echo $name; ?></p>

<p>Email: <?php echo $email; ?></p>

<p>Password: <?php echo $password; ?></p>

<p>Date of Birth: <?php echo $dob; ?></p>

</body>

</html>

1. Viết tập tin sinhvien\_class.php để khai báo 1 lớp sinhvien với các thuộc tính: mssv, hoten, ngaysinh với kiểu giá trị phù hợp. Viết các phương thức xây dựng, hủy, gán trị, trả về giá trị của các thuộc tính, viết hàm tính tuổi. Khai báo 1 biến sv có kiểu class sinhvien, gán các giá trị cần thiết với thông tin của bạn và thử tính tuổi dựa vào hàm vừa mới xây dựng trong class.

**<?php**

**class SinhVien {**

**// Các thuộc tính của lớp**

**public $mssv;**

**public $hoten;**

**public $ngaysinh;**

**// Phương thức khởi tạo (constructor)**

**public function \_\_construct($mssv, $hoten, $ngaysinh) {**

**$this->mssv = $mssv;**

**$this->hoten = $hoten;**

**$this->ngaysinh = $ngaysinh;**

**}**

**// Phương thức hủy (destructor)**

**public function \_\_destruct() {**

**echo " => Xong!";**

**}**

**// Phương thức gán giá trị cho mssv**

**public function setMSSV($mssv) {**

**$this->mssv = $mssv;**

**}**

**// Phương thức gán giá trị cho hoten**

**public function setHoTen($hoten) {**

**$this->hoten = $hoten;**

**}**

**// Phương thức gán giá trị cho ngaysinh**

**public function setNgaySinh($ngaysinh) {**

**$this->ngaysinh = $ngaysinh;**

**}**

**// Phương thức trả về giá trị của mssv**

**public function getMSSV() {**

**return $this->mssv;**

**}**

**// Phương thức trả về giá trị của hoten**

**public function getHoTen() {**

**return $this->hoten;**

**}**

**// Phương thức trả về giá trị của ngaysinh**

**public function getNgaySinh() {**

**return $this->ngaysinh;**

**}**

**// Phương thức tính tuổi**

**public function tinhTuoi() {**

**$now = new DateTime();**

**$ngaysinh = new DateTime($this->ngaysinh);**

**$diff = $now->diff($ngaysinh);**

**return $diff->y;**

**}**

**}**

**// Sử dụng lớp SinhVien**

**$sv = new SinhVien("B2110044", "Ngo Nhu Huynh", "2003-04-21");**

**// Hiển thị thông tin sinh viên**

**echo "MSSV: " . $sv->getMSSV() . "<br>";**

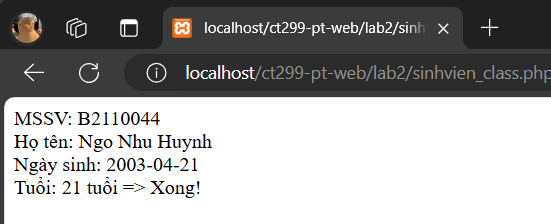
**echo "Họ tên: " . $sv->getHoTen() . "<br>";**

**echo "Ngày sinh: " . $sv->getNgaySinh() . "<br>";**

**// Tính tuổi và hiển thị**

**echo "Tuổi: " . $sv->tinhTuoi() . " tuổi";**

* **?>**



**Chú ý:**

* Các bạn nộp file word: Quy tắc đặt tên file: **<mssv>-<hoten>-<bai><stt\_bai thực hành>.docx** nộp lên Classroom (VD: **B123456-NguyenVanA-bai1.docx**), kèm với các file khác được yêu cầu như phần câu hỏi đã nêu. **Ngoại trừ file word trả lời câu hỏi, các file còn lại các bạn nén vào 1 file zip**. File zip đặt tên như file word.
* Mỗi câu các bạn trả lời bằng hình hoặc dạng text tùy vào yêu cầu của câu hỏi và **TRẢ LỜI THEO ĐÚNG THỨ TỰ CÂU HỎI**. Nếu câu nào không trả lời được các bạn cứ để số thứ tự câu hỏi và bỏ trống phần trả lời.
* Các câu trả lời có tham khảo trên Internet phải trích dẫn link/nguồn.
* *Vi phạm 1 trong các điều sau đây bài thực hành sẽ bị 0 điểm:*

○Đặt tên KHÔNG ĐÚNG quy tắc được yêu cầu.

○ Bài không đủ các thành phần (word, code+data (nếu có),...) đã được yêu cầu.

○ Bài không thực hiện đúng yêu cầu “**Ngoại trừ file word trả lời câu hỏi, các file còn lại các bạn nén vào 1 file .zip”**

○ Bị phát hiện copy, sao chép từ các bạn khác

○ Phần trả lời không ghi rõ trả lời cho câu nào

○ Thứ tự câu trả lời không đúng thứ tự câu hỏi

1. https://www.w3schools.com/php/php\_intro.asp [↑](#footnote-ref-1)
2. [↑](#footnote-ref-2)