**BÀI TẬP THỰC HÀNH 02**

**LÀM QUEN VỚI NGÔN NGỮ PHP**

MSSV: B2103453

HoTen: Võ Đoàn Ngọc Châu

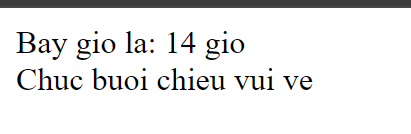
1. Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.

*- Tạo tập tin hello.php hiển thị chữ “Hello!”*



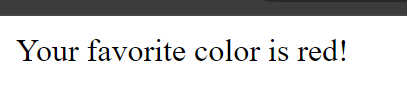
*- Tạo cấu trúc if, elseif, và else xét biết t có kiểu giờ hiện hành, và xét*

*theo giờ hiện để hiện thị thông báo*

**

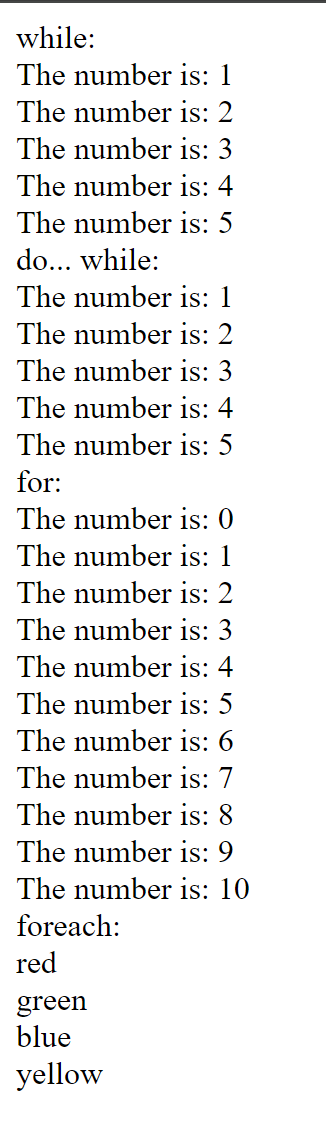
*- Tạo cấu trúc switch xét biết favcolor có chứa giá trị gì để hiển thị*

*thông báo*

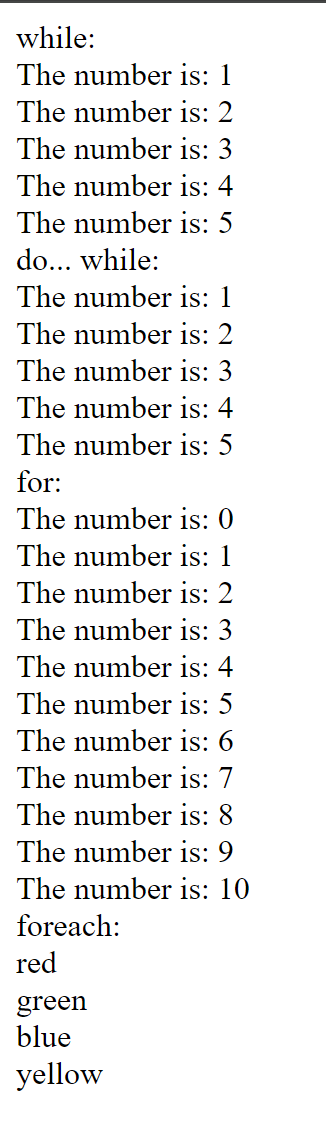
**

*- Tạo file loop.php viết các lệnh và quan sát kết quả*

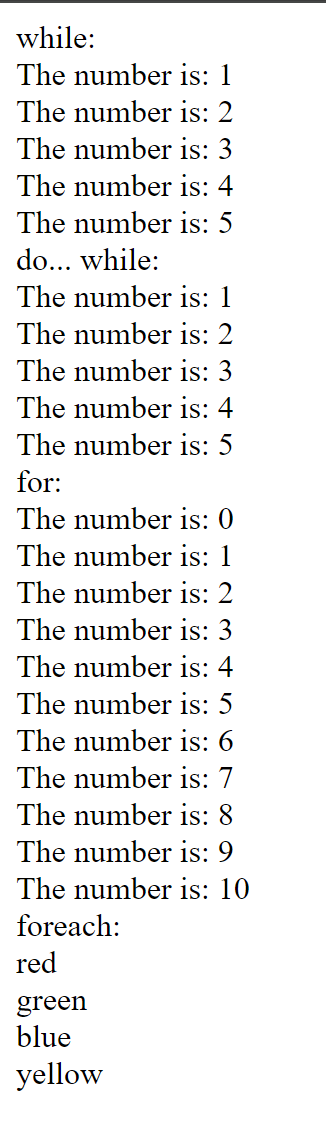
+ Lệnh while: x lặp từ 1 đến 5 và in kết quả ra màn hình

**

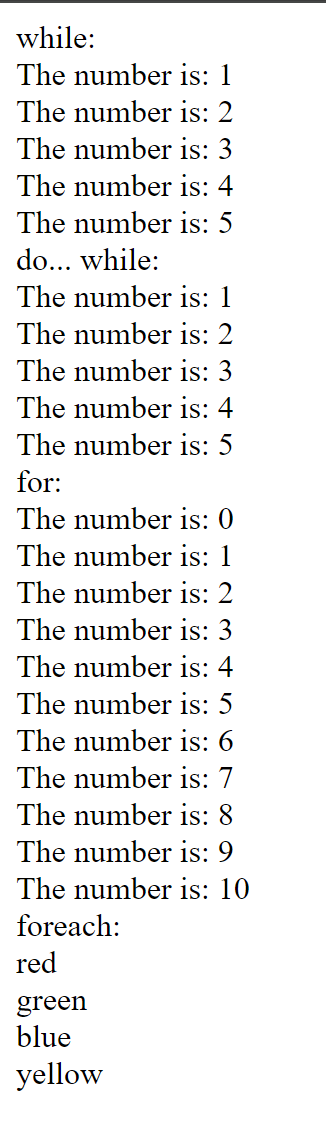
*+ Lệnh do…while: x lặp từ 1 đến 5 và in kết quả ra màn hình*

**

*+ Lệnh for: x lặp từ 0 đến 10 và in kết quản ra màn hình*

**

*+ Lệnh foreach: lặp qua từng phần tử của mảng colors và in ra màn hình*

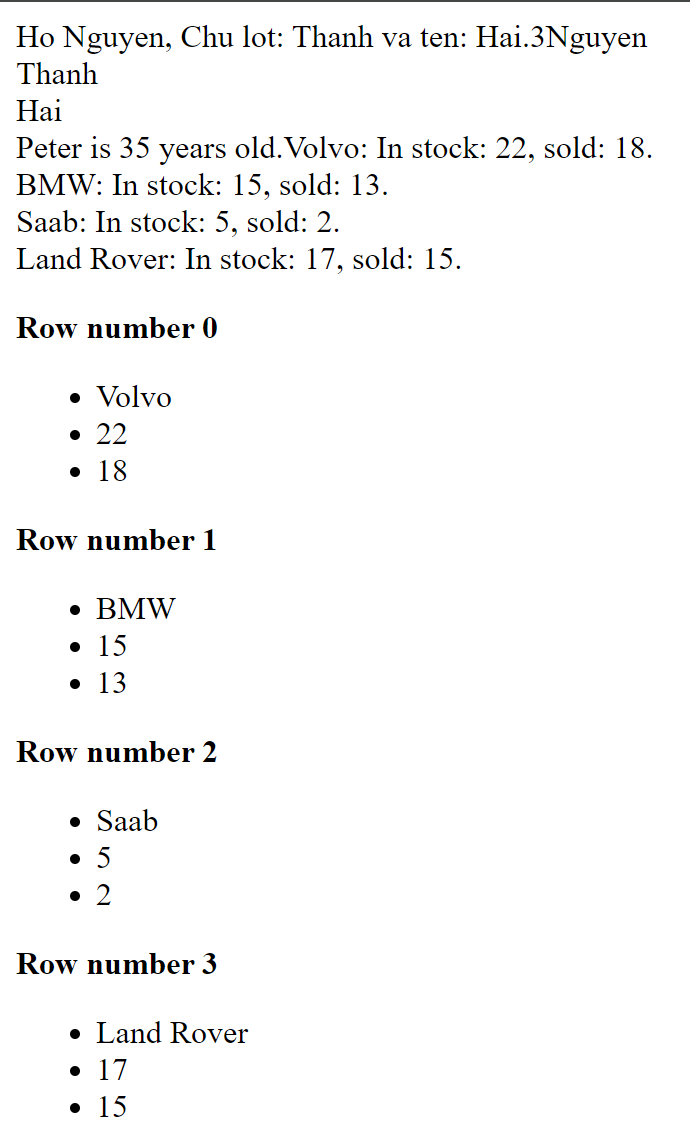
**

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

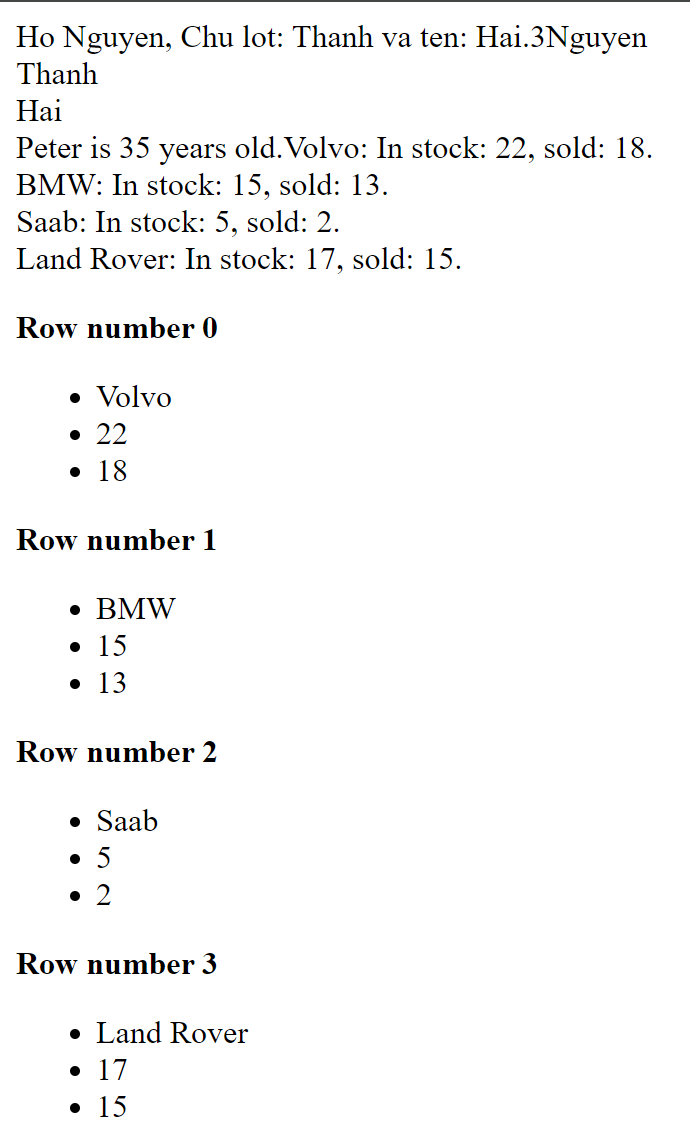
**

* Mảng:

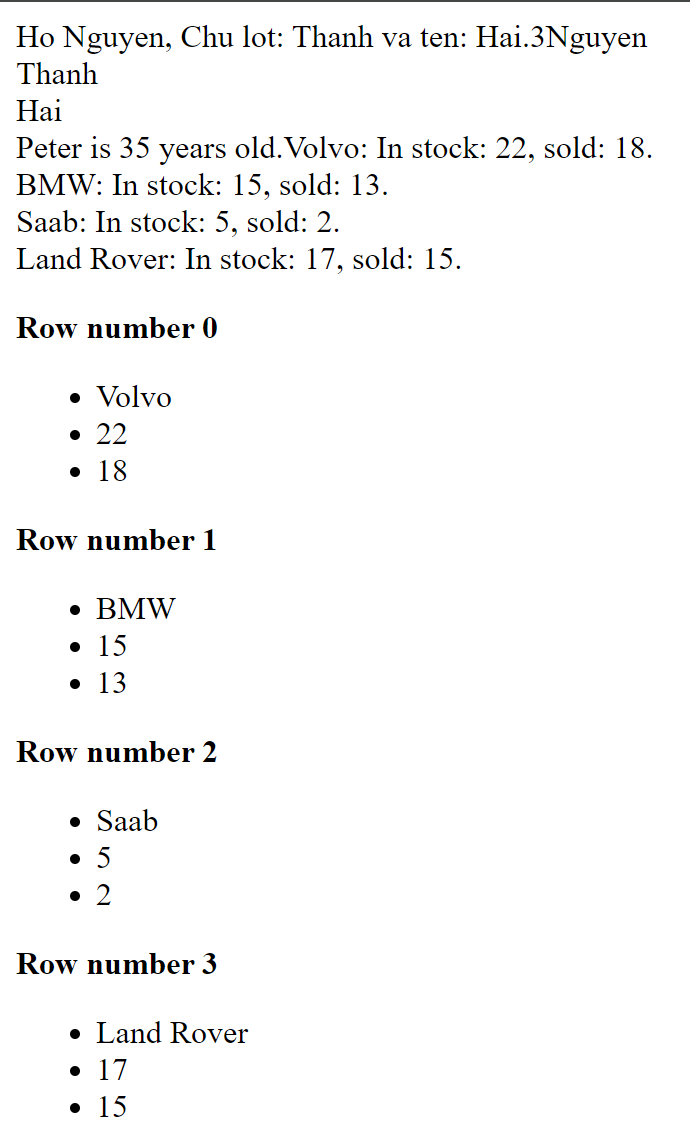
+ Sử dụng mảng và in giá trị

**

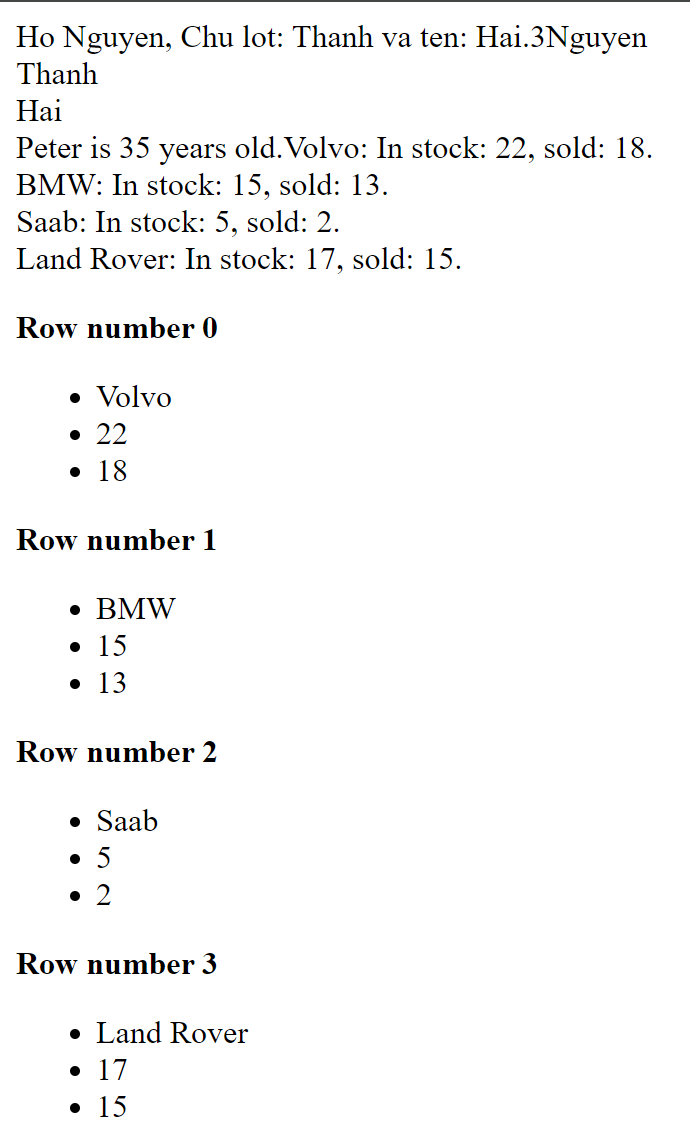
+ Mảng “gom key => giá trị”



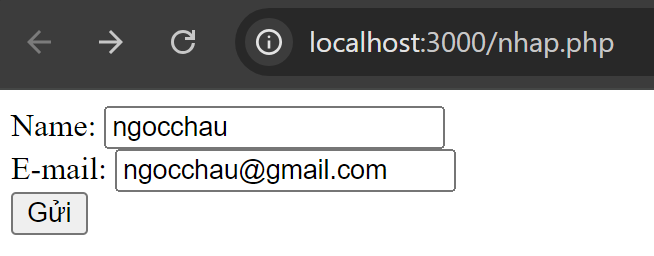
+ Mảng đa chiều

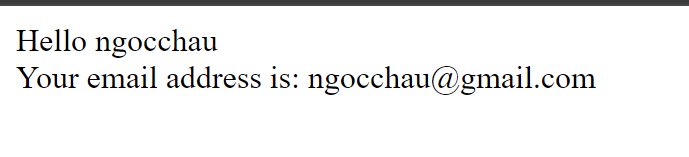


+ Hiển thị mảng đa chiều qua for

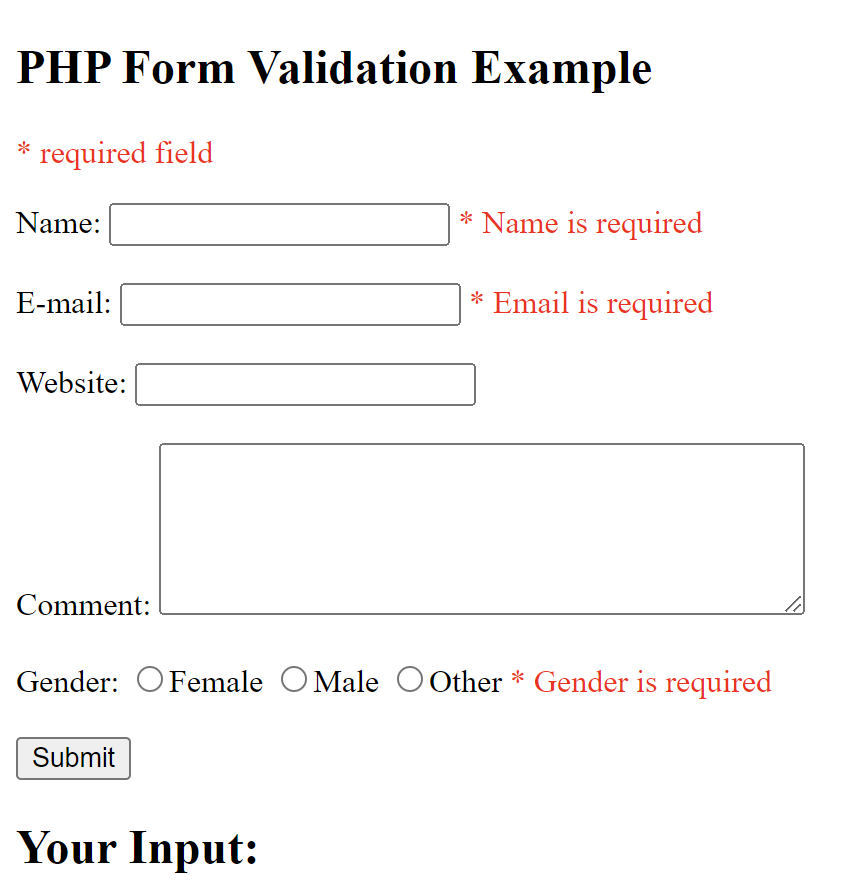


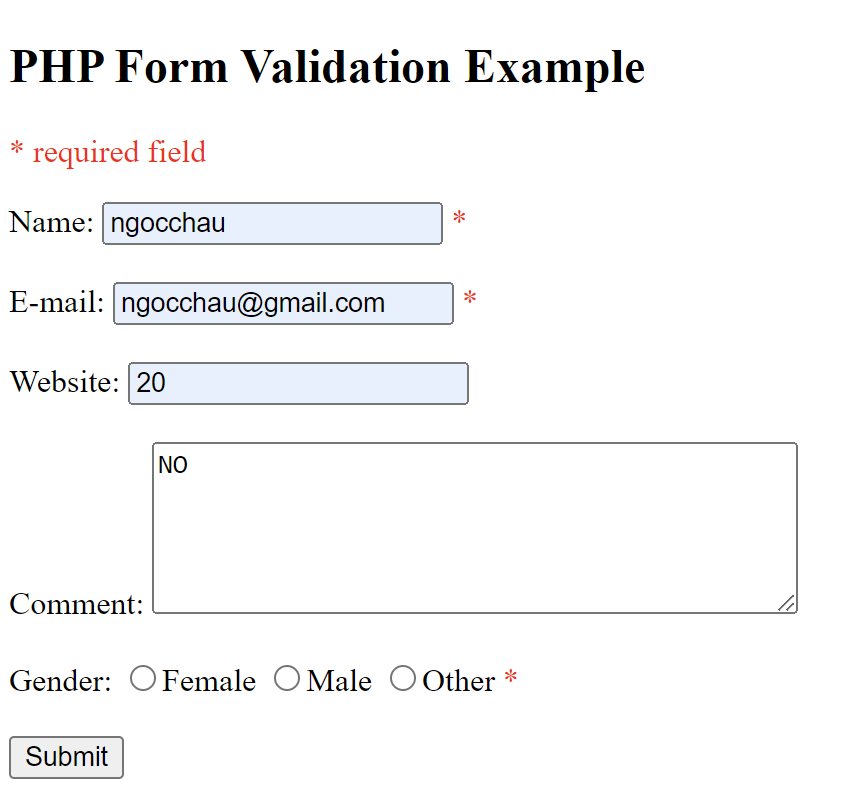
* Xử lí form:





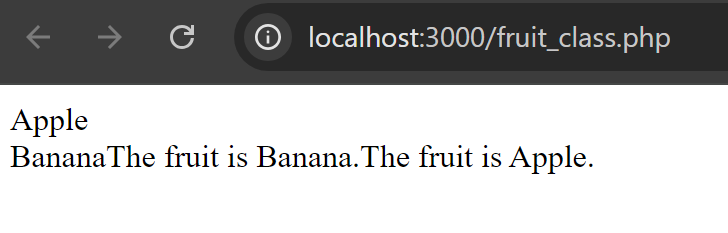
* Kiểm tra dữ liệu trên form:







*- Hướng đối tượng (OOP) trong PHP*

**

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:

Khai báo biến bằng ký tự $ đứng trước tên biến.

Tên biến phải là một chứ cái hoặc dấu gạch dưới\_, không bắt đầu bằng chữ số.

Tên biến chứa chữ cái, số và dấu gạch dưới, không chứa kí tự đặt biệt, khoảng trắng.

Tên biến phân biệt chữ hoa, thường.

Ví dụ: $ten = “Nguyễn Văn A”; $\_so\_luong = 20; $tuoi\_2 = 20;

* Gán giá trị cho biến:

Sử dụng toán tử gán = để gán giá trị cho biến.

+ Kiểu chuỗi: $ten = “Nguyễn Văn A”;

+ Kiểu số: $tuoi = 21;

+Kiểu ngày tháng: $ngay\_sinh = “2003-02-03”;

* Chú ý khi đặt tên biến :

+ Tên biến không trùng với từ khóa đặt biệt trong PHP. Ví dụ: echo, class…

+Tên biến nên có ý nghĩa và dễ hiểu.

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

- Ký hiệu bắt đầu khối lệnh: <?php

- Ký hiệu kết thúc khối lệnh PHP: ?>

Ví dụ:

<?php

//Bắt đầu khối lệnh

echo” Xin chào !”;

// Kết thúc khối lệnh

?>

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

- Comment trên một dòng: sử dụng ký hiệu //

- Comment trên nhiều dòng: sử dụng ký hiệu /\*\*/

Ví dụ:

+ Comment trên 1 dòng:

<?php

//Bắt đầu khối lệnh

echo” Xin chào !”;

// Kết thúc khối lệnh

?>

+ Comment trên nhiều dòng:

<?php

/\* Đây là comment trên nhiều dòng

Dòng thứ 2 của comment \*/

echo” Xin chào !”;

?>

5. 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.

- Có 4 loại khối lệnh trong vòng lặp:

+ **for**: Thực hiện vòng lặp với điều kiện kiểm tra và sửa đổi biến đếm.

*Ví dụ:* for($i =0; $i < 10; $i++) {

echo $i;

}

+ **while**: Vòng lặp chạy miễn điều kiện là đúng.

*Ví dụ:* $i = 0;

while ($i < 10) {

echo $i;

$i++;

}

**+ do…while**: Vòng lặp chạy ít nhất một lần, sau đó tiếp tục nếu điều kiện đúng.

*Ví dụ:* $i = 0;

do {

echo $i;

$i++;

}while ($i < 10);

+ **foreach**: Vòng lặp dành riêng để lặp qua các phần tử của mảng hoặc đối tượng.

*Ví dụ:* $mang = array("Táo", "Cam", "Chuối");

foreach ($mang as $trai\_cay) {

echo $trai\_cay;

}

- Sự khác biệt của foreach với các loại khối lệnh vòng lặp khác:

foreach chỉ được sử dụng để lặp qua mảng hoặc đối tượng. giúp đơn giản hóa việc lặp qua từng phần tử trong mảng, không cần phải biết kích thước của mảng hoặc tạo biến đếm như với vòng for.

- Cách dùng của lệnh “continue” và lệnh “break” trong các vòng lặp:

**continue**: Bỏ qua các lệnh còn lại trong vòng lặp hiện tại và chuyển sang lần lặp kế tiếp.

Ví dụ:

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

if ($i == 2) {

continue; // Bỏ qua khi $i = 2 }

echo $i; // In ra: 0, 1, 3, 4

}

**break**: Dừng vòng lặp hoàn toàn, thoát khỏi vòng lặp ngay lập tức.

Ví dụ:

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

if ($i == 3) {

break; // Thoát khỏi vòng lặp khi $i = 3

}

echo $i; // In ra: 0, 1, 2

}

6. 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);

            }

            $number = 15;

            $giaithua = tinhGiaiThua($number);

echo "$number giai thua: $giaithua";

?>



7. 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) {

    $lenA = count($a);

  $lenB = count($b);

    if ($lenA != $lenB) {

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

        return "Lỗi: Hai mảng không 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);

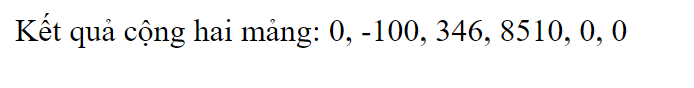
} else {

    echo $result;

}

?>

- Khi số phần tử bằng nhau:



- Khi số phần tử không bằng nhau:



8. Chạy ví dụ tập tin validation.php, dựa vào link 2, bạn hãy giải thích những bước diễn ra quá trình này. <https://www.w3schools.com/php/php_form_validation.asp>

- Biến và hàm khởi tạo:

+ $name, $email,$gender,$comment, $website khởi tạo để lưu trữ các dữ liệu nhập vào.

+ Hàm test\_input xử lý dữ liệu đầu vào và thực hiện các bước:

trim: bỏ khoảng trắng ở đầu chuỗi và cuối chuỗi.

stripslashes: bỏ kí tự backslashes từ chuỗi

htmlspecialchars: Chuyển ký tự đặc biệt thành thực thể HTML.

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

+ $\_SERVER[“REQUEST\_METHOD”] kiểm tra biểu mẫu gửi đi hay chưa

+ Khi biểu mẫu được gửi đi, lấy dữ liệu từ trường của biểu mẫu và sử dụng hàm test\_input xử lý dữ liệu

* Biểu mẫu HTML:

+ Tạo biểu mẫu HTML với các trường nhập dữ liệu như tên, email, website, comment, và gender.

+ Sử dụng thẻ **<input>, <textarea>, và <select>** tạo các trường nhập dữ liệu và ô lựa chọn.

+ Đặt thuộc tính **method="post"** để biểu mẫu gửi dữ liệu bằng phương thức POST.

+ Đặt thuộc tính **action="<?php echo htmlspecialchars($\_SERVER["PHP\_SELF"]);?>"** để biểu mẫu gửi dữ liệu đến chính trang hiện tại.

* Hiển thị kết quả:

+ Sau khi biểu mẫu gửi đi, hiển thị thông báo nhập vào trên trang.

+ Sử dụng echo hiển thị thông tin về tên, email, comment, website và gender.

9. 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.

Tạo tập tin nhap2.php

<!DOCTYPE HTML>

<html>

<body>

<form action="welcome2.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>

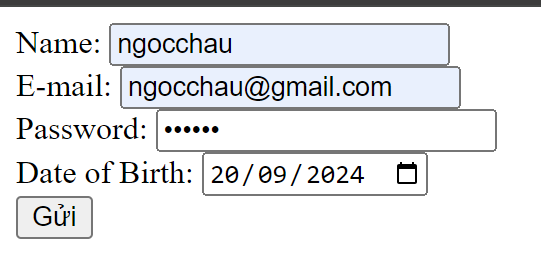
        Date of Birth: <input type="date" name="dob"><br>

        <input type="submit">

</form>

</body>

</html>



- Tạo tập tin 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>

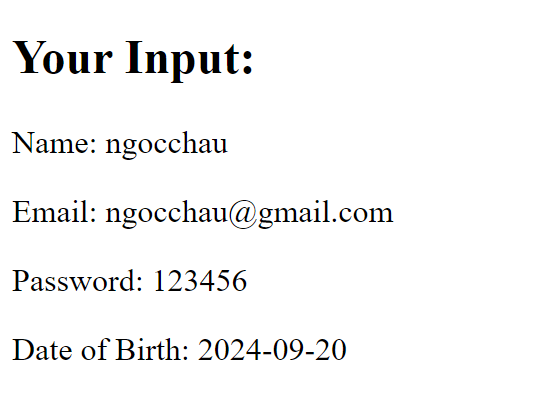
<p>Email: <?php echo $email; ?></p>

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

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

</body>

</html>



10. 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 " => Ket thuc!";

    }

        // 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("B2103453", "Vo Doan Ngoc Chau", "2003-02-30");

// Hiển thị thông tin sinh viên

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

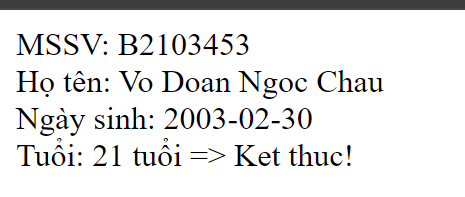
echo "Họ tên: " . $sv->getHoTen() . "<br>";

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

// Tính tuổi và hiển thị

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

?>



**TÀI LIỆU THAM KHẢO:**

**[1]** [**https://www.w3schools.com/php**](https://www.w3schools.com/php)