BÀI 2: LẬP TRÌNH PHP CƠ BẢN (TT)

Php & Form

Mục TIÊU

- PHP Include Files
- Dates
- PHP Variables Scope
- Variable testing
- PHP và Form
- **⊙**GET & POST
- REQUEST

PHP INCLUDE FILES

- ☐ The include (or require) statement takes all the text/code/markup that exists in the specified file and copies it into the file that uses the include statement.
- □Including files is very useful when you want to include the same PHP, HTML, or text on multiple pages of a website

Syntax

include 'filename';

or

require 'filename';

PHP INCLUDE FILES

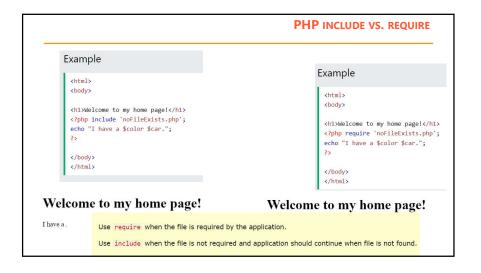
□PHP include and require Statements

- The include and require statements are identical, except upon failure
- require will produce a fatal error (E_COMPILE_ERROR) and stop the script
- ❖include will only produce a warning (E_WARNING) and the script will continue

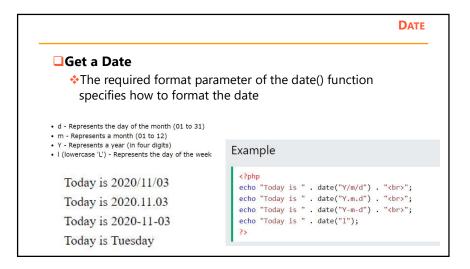




PHP INCLUDE EXAMPLES Assume we have a file called "vars.php", with some variables defined: Example <?php <html> \$color='red'; <body> \$car='BMW'; ?> <h1>Welcome to my home page!</h1> <?php include 'vars.php';</pre> echo "I have a \$color \$car."; </body> </html> Welcome to my home page! I have a red BMW.



The PHP Date() Function ❖The PHP date() function formats a timestamp to a more readable date and time. Syntax date(format, timestamp) Parameter Description format Required. Specifies the format of the timestamp timestamp Optional. Specifies a timestamp. Default is the current date and time



```
Here are some characters that are commonly used for times:

• H - 24-hour format of an hour (00 to 23)
• h - 12-hour format of an hour with leading zeros (01 to 12)
• i - Minutes with leading zeros (00 to 59)
• s - Seconds with leading zeros (00 to 59)
• a - Lowercase Ante meridiem and Post meridiem (am or pm)

Example

**Replace**

The time is 09:41:12am

**Replace**

The time is 09:41:12am
```

```
□ date('Y-m-d H:i:s'): ngày hiện hành.
□ date('Y-m-d H:i:s', time() - 60 * 60 * 24): ngày hôm qua
□ date('F j Y, H:i:s'): các định dạng khác của ngày tháng năm (Tham khảo thêm https://www.php.net/manual/en/function.date.php)
□ time(): thời gian hiện hành
```

DATES

- □date_parse: thông tin cụ thể như năm, tháng, ngày, giờ, phút, giây phục vụ cho quá trình tính toán
 - ❖Input: \$dateString: Chuỗi thông tin ngày tháng cần lấy thông tin
 - Output: Trả về mảng chứa thông tin chi tiết về thời gian đã chọn

```
// Parse date: https://www.php.net/manual/en/function.date-parse.php
$dateString = '2020-02-06 12:45:35';
$parsedDate = date_parse($dateString);
echo '';
var_dump($parsedDate);
echo '';
```

DATES

- □ date_parse_from_format: lấy thông tin về ngày cụ thể được truyền vào căn cứ vào kiểu định dạng của chuỗi thời gian.
 - Input:

```
$format: Định dạng
```

\$dateString: Chuỗi thời gian cần lấy thông tin

❖Output: Trả về mảng chứa thông tin chi tiết về thời gian đã chọn

```
$dateString = 'February 4 2020 12:45:35';

$parsedDate = date_parse_from_format('F j Y H:i:s', $dateString);
echo '';
var_dump($parsedDate);
echo '';
```

PHP VARIABLES SCOPE

- 1. local
- 2. global
- 3. static

1. GLOBAL

2. LOCAL SCOPE

```
function myTest() {
    $x = 5; // local scope
    echo "Yariable x inside function is: $x";
}
myTest();

// using x outside the function will generate an error echo "Yariable x outside function is: $x";
}
```

3. PHP THE GLOBAL KEYWORD

The global keyword is used to access a global variable from within a function

```
$\text{?php}
$x = 5;
$y = 10;

function myTest() {
    global $x, $y;
    $y = $x + $y;
}

myTest();
echo $y; // outputs 15
}
```

PHP THE STATIC KEYWORD

```
<?php
function myTest() {
    static $x = 0;
    echo $x;
    $x++;
}

myTest();
myTest();
myTest();
?>
```

when a function is completed/executed, all of its variables are deleted. However, sometimes we want a local variable NOT to be deleted. We need it for a further job.

GLOBAL SCOPE

- SGLOBALS: được sử dụng để truy xuất global variables từ bất kỳ đâu trong PHP script
- □PHP lưu trữ tất cả các global variables trong mảng gọi là \$GLOBALS [index] với index là tên của biến.
- ■Ví du

VARIABLE TESTING

- ☐ PHP xử lý dữ liệu do người dùng cung cấp. Dữ liệu cần được kiểm tra trước khi sử dụng để xác nhận rằng nó tồn tại và có giá trị hợp lệ. PHP cung cấp một số cấu trúc dựng sẵn được sử dụng cho mục đích này.
- □ Isset: trả về true nếu biến tồn tại và đã được gán một giá trị khác null

```
isset($a); // false
$a = 10;
isset($a); // true
$a = null;
isset($a); // false
```

VARIABLE TESTING

- □ Empty: kiểm tra xem biến được chỉ định có giá trị rỗng (empty) hay không như là null, 0, false hoặc một chuỗi rỗng và trả về true nếu đúng.
- □ Empty: cũng trả về true nếu biến không tồn tại.

```
empty($b); // true
$b = false;
empty($b); // true
```

VARIABLE TESTING

□ Is_null : được sử dụng để kiểm tra xem biến có được đặt là null hay không

```
// Prior to PHP 8
echo is_null($d); // true (undefined variable notice)
// As of PHP 8
echo is_null($d); // error (TypeError)
```

VARIABLE TESTING

unset: xoá biến khỏi phạm vi hiện tại.

```
$var = null; // free memory
unset($var); // delete variable
```

□ Null Coalescing Operator(??) là một dạng ngắn gọn của sử dụng toán tử bậc 3 với isset

```
$name = isset($x) ? $x : 'unknown';
$x = null;
$name = $x ?? 'unknown'; // "unknown"
```

VARIABLE TESTING

□ Null coalescing assignment operator(??=) : gán giá trị cho một biến chỉ khi biến đó chưa được gán (null). Nó cũng có thể được sử dụng với các biến không xác định.

```
// Assign value if $name is unassigned
$name ??= 'unknown';

// Same as above
if(!isset($name)) { $name = 'unknown'; }
```

VARIABLE TESTING Mô tả Tên hàm Is array() True nếu biến là 1 mảng Is bool() True nếu biến là 1 bool Is callable() True nếu biến có thể được gọi như 1 hàm Is_float(), is_double(), is_real() True nếu biến là 1 float Is int(), is integer(), is long() True nếu biến là 1 integer Is null() True nếu biến được thiết lập là null Is numeric() True nếu biến là số hoặc chuỗi số Is object() True nếu biến là 1 object Is string() True nếu biến là 1 chuỗi

VARIABLE TESTING

- ☐ Variable information: PHP xây dựng sẵn 3 hàm cho việc nhận thông tin về biến: print_r, var_dump, và var_export
- print_r hiển thị giá trị của biến theo cách mà con người có thể đọc được. Rất hữu ích cho mục đích debug.

```
$a = array('one', 'two', 'three');
print_r($a);
```

□Output với lệnh print

Array ([0] => one [1] => two [2] => three)

□Output với lệnh var_dump

```
array(3) {
  [0]=> string(3) "one"
  [1]=> string(3) "two"
  [2]=> string(5) "three"
```

PHP & FORM

- ☐ HTML form có 2 thuộc tính bắt buộc: action và method
- Thuộc tính action chỉ định tập tin mà dữ liệu từ form được gửi đến khi form submit.
- ☐ Ví dụ phía dưới đây: form sẽ gửi input myString đến mypage.php khi form submit

</html>

PHP & FORM

- ☐ Thuộc tính bắt buộc thứ 2 của form là phương thức gửi, có 2 loại GET và POST
- ☐Gửi dữ liêu với POST
 - ❖Dữ liệu được gửi sẽ không hiển thị trên thanh url.
 - Khi form được submit, trình duyệt tải trang mypage.php và gửi dữ liệu của form.
 - ❖Dữ liệu sẽ được nhận thông qua mảng \$_POST. Tên của thuộc tính ở đầu vào của form thành khoá(key) trong mảng kết hợp.

```
<?php // mypage.php ?>
<!doctype html>
<html>
<body>
<?php echo $_POST['myString']; ?>
</body>
</html>
```

```
☐ Gửi dữ liệu với GET

Phương thức GET sẽ gửi dữ lieu trong url

Dữ liệu gửi đi sẽ được nhận thông qua mảng $_GET

// mypage.php
echo $_GET['myString'];

Cách đọc các GET variable từ URL

http://localhost/PHP1/display.php?input text=hello&input name=nona
me@gmail.com

$id_text = $_GET['input_text'];
$in_email = $_GET['input_email**];
```

PHP & FORM ■Ví du en display.php m index.php X m index.php and display.php <form action="display.php" method="GET"> <?php <label for="user">User = \$_GET['input_text']; <input type="text" name="input_text"> \$id text = \$_GET['input_email']; </label> \$in_email <label for="user">Email <input type="email" name="input_email"> </label> <input type="submit" /> </form>

PHP & FORM

- ☐ Vì dữ liệu được chứa trong thanh địa chỉ, các biến không chỉ được chuyển qua form của HTML mà còn thông qua liên kết HTML => cách để chuyển biến từ trang này sang trang khác.
- □Ví du

```
<?php // sender.php ?>
<!doctype html>
<html>
<body>
    <a href="receiver.php?myString=Foo+Bar">link</a>
</body>
</html>
```

PHP & FORM

```
<?php // receiver.php ?>
<!doctype html>
```

■Ví du

</html>

```
<html>
<body>
<?php echo $_GET['myString']; // "Foo Bar" ?>
</body>
```

REQUEST ARRAY

- ☐ Nếu việc gửi dữ liệu với \$_POST, \$_GET không quan trọng thì có thể sử dụng \$ REQUEST.
- ☐ Mảng \$_REQUEST chứa cả \$ GET và \$ POST. Ví dụ

REQUEST ARRAY

☐ Việc chỉnh sửa các thành phần của \$_POST hoặc \$_GET hoàn toàn không ảnh hưởng đến các thành phần của \$ REQUEST

```
$_GET['foo'] = 'a';

$_POST['bar'] = 'b';

var_dump($_GET); // Element 'foo' is string(1) "a"

var_dump($_POST); // Element 'bar' is string(1) "b"

var_dump($_REQUEST); // Does not contain elements 'foo' or 'bar'

?>
```

PHP & FORM

- ☐ Mối quan tâm về bảo mật:
 - Mọi dữ liệu do người dùng cung cấp đều có thể bị thao túng; Do đó cần được xác nhận và cải thiện (sanitized) trước khi sử dụng.
 - Validation (Xác thực) có nghĩa là đảm bảo rằng dữ liệu cần đúng hình thức mình muốn như kiểm dữ liệu, phạm vi, nội dung.
 - ❖Ví dụ: xác thực địa chỉ email

```
if(!filter_var($_POST['email'], FILTER_VALIDATE_EMAIL))
echo "Invalid email address";
```

PHP & FORM

- ☐ Mối quan tâm về bảo mật:
 - Dữ liệu do người dùng cung cấp dưỡi dạng text, httmlspecialchars sẽ được sử dụng.
 - Htmlspecialchars sẽ vô hiệu hoá bất kì đánh dấu HTML nào để đầu vào của người dùng được hiển thị nhưng không được diễn giải

```
// Sanitize for web page use
echo htmlspecialchars($_POST['comment']);
```

PHP & FORM

■ Submitting array

- Dữ liệu của form có thể nhóm thành mảng bằng cách sử dụng dấu ngoặc vuông [] phía sau tên biến trong form.
- ❖Việc tạo mảng này có thể thực hiện đối với các input của form gồn <input>, <select> và <textarea>

```
<input type="text" name="myArr[]">
<input type="text" name="myArr[]">
```

❖Có thể tạo key cho mảng các input

<input type="text" name="myArr[name]">

PHP & FORM

■Submitting array

Lấy mảng dữ liệu được gửi từ form.



SUMARRY

- ✓ Dates
- ✓ Variable testing
- ☑PHP và Form
- **☑**GET & POST
- **☑** REQUEST