

# PHP for Base

Giảng viên: Bùi Quang Đăng

# Contents



Working with session, cookies



Files in PHP



FileUpload in PHP

# PHP for Base

## Session, Cookies

# PHP for Base

## ❖ Session

- Là một biến trong PHP sử dụng để lưu trữ hoặc thay đổi thiết lập trong một phiên làm việc của người dùng.
- Biến này có thể sử dụng trên tất cả trên các trang của một ứng dụng
- Một **session** được bắt đầu khi người dùng truy cập ứng dụng trên trình duyệt (web browser) mới và kết thúc khi người dùng tắt trình duyệt hoặc **timeout**.
- **Session** sẽ tạo ra một unique id (uid) duy nhất để lưu trữ khi người dùng ghé thăm.

# PHP for Base

## ❖ Session

- Để sử dụng Session trong PHP người lập trình cần phải kích hoạt phiên làm việc thông qua phương thức **session\_start()**.
- Cú pháp:

**`$_SESSION['Ten_bien'] = Gia_tri_can_luu`**

Trong đó:

- **Ten\_bien**: Do người dùng đặt mô tả được ý nghĩa tùy theo mục đích sử dụng của mình.
- **Gia\_tri\_can\_luu**: Là thông tin muốn lưu trữ vào session trong PHP

# PHP for Base

## ❖ Session

- Sử dụng phương thức **isset()** để kiểm tra 1 biến session có tồn tại hay không.
- Sử dụng phương thức **unset()** để xóa 1 biến session trong PHP.

Ví dụ:

```
<?php
session_start();

if(isset($_SESSION['useronline']))
    $_SESSION['useronline']=$_SESSION['useronline']+1;
else
    $_SESSION['useronline']=1;
echo "Số người đang online là: ". $_SESSION['useronline'];
?>
```

# PHP for Base

## ❖ Session

- Sử dụng phương thức **session\_destroy()** để xóa các session đã lưu trong php.

```
session_start();  
$timeout = 60; // Number of seconds until it times out.  
  
if(isset($_SESSION['timeout'])) {  
  
    $duration = time() - (int)$_SESSION['timeout'];  
  
    if($duration > $timeout) {  
        session_destroy(); // Destroy the session and restart it.  
  
        session_start();  
    }  
}  
$_SESSION['timeout'] = time(); // Update the timeout field with the current time.
```

# PHP for Base

## ❖ Cookies

- Một **cookie** thường sử dụng để nhận dạng người dùng truy cập ứng dụng, hệ thống.
- Nó là một file nhỏ được server nhúng vào máy tính của người dùng được sử dụng để đọc, ghi thông tin thông qua trình duyệt khi cần.
- Trong PHP chúng ta có thể đọc và lấy lại thông tin được lưu trữ trong **Cookie**.



# PHP for Base

## ❖ Cookies

- Trong PHP để tạo ra một cookie người ta có thể sử dụng phương thức **setcookie()** và lấy thông tin qua biến **\$\_COOKIE**.
- **Cú pháp:**
  - **setcookie(name, value, expire, path, domain)**

Trong đó:

- **name:** là tên của cookie do người dung đặt
- **value:** giá trị cần lưu trữ
- **expire:** Thời gian lưu trữ của cookie
- **path, domain:** đường dẫn và domain lưu trữ cookie

# PHP for Base

## ❖ Cookies

- Trong PHP để tạo ra một cookie người ta có thể sử dụng phương thức **setcookie()** và lấy thông tin qua biến **\$\_COOKIE**.

Ví dụ:

```
<?php
```

```
$expire=time()+60*60*24*30;
```

```
setcookie("useronline", "dangbq-stanford", $expire);
```

```
?>
```

# PHP for Base

## ❖ Cookies

- Trong PHP để lấy thông tin qua biến **\$\_COOKIE**.
- Sử dụng phương thức **print\_r(\$\_COOKIE)** để lấy tất cả các thông tin về cookie.
- **Cú pháp:**

**\$\_COOKIE["Ten\_bien"]**

Ví dụ:

```
<?php
// Print a cookie
echo $_COOKIE["useronline"];

// A way to view all cookies
print_r($_COOKIE);
?>
```

# PHP for Base

## ❖ Cookies

- Trong PHP để xóa một cookie người ta sẽ sử dụng phương thức **setcookie** trong đó tham số expire sẽ được thiết lập là trừ đi chính số thời gian được tồn tại cookie trước đó.

Ví dụ:

```
<?php
```

```
$expire=time()- 60*60*24*30;
```

```
setcookie("useronline", "dangbq-stanford", $expire);
```

```
?>
```

# PHP for Base

## Working files in PHP

# PHP for Base

## ❖ File in PHP

- Sử dụng để làm việc với các file trong PHP
- Có thể đọc, ghi thông tin trên file
- Sử dụng các phương thức được cung cấp trong PHP để thao tác và làm việc với file tùy theo từng trường hợp cụ thể

# PHP for Base

## ❖ File in PHP

- **fopen()** là phương thức sử dụng để mở file hoặc Url trong PHP
- Cú pháp: **fopen**(filename, filemode)
  - Trong đó:
    - **filename**: là tên file cần mở
    - **filemode**: kiểu file sử dụng để mở như đọc, ghi,...

Ví dụ:

```
$file=fopen("welcome.txt","r") or exit("Khong mo duoc file");
```

# PHP for Base

## ❖ File in PHP

- Các kiểu mở file trong PHP khi làm việc với file như sau:

Ký hiệu	Mô tả
<b>r</b>	Chỉ đọc file
<b>r+</b>	Có khả năng đọc/ghi file
<b>w</b>	Chỉ ghi file. Mở file và xóa nội dung trong file hoặc tạo ra một file mới nếu không tồn tại.
<b>w+</b>	Có khả năng đọc/ghi file. Mở file và xóa nội dung trong file hoặc tạo ra một file mới nếu không tồn tại.
<b>a</b>	Thêm nội dung vào file. Mở và ghi file nếu tồn tại hoặc tạo một file mới.



# PHP for Base

## ❖ File in PHP

- Các kiểu mở file trong PHP khi làm việc với file như sau:

Ký hiệu	Mô tả
<b>a+</b>	Đọc / thêm vào file. Giữ nguyên nội dung đã có và chèn thêm vào cuối của file.
<b>x</b>	Chỉ ghi file, tạo ra một file mới và trả về giá trị FALSE, lỗi nếu file đã tồn tại
<b>x+</b>	Đọc và ghi file, tạo ra một file mới và trả về giá trị FALSE, lỗi nếu file đã tồn tại.

# PHP for Base

## ❖ File in PHP

- **fclose()** là phương thức sử dụng để đóng một file mở trong PHP
- Cú pháp: **fclose**(\$file\_can\_dong)

Ví dụ:

```
$file=fopen("welcome.txt","r") or exit("Khong mo duoc file");
```

```
fclose($file);
```

# PHP for Base

## ❖ File in PHP

- **feof()** là phương thức sử dụng để kiểm tra kết thúc file hay chưa trong PHP
- Cú pháp: **feof**(\$file\_can\_dong)

Ví dụ:

```
$file=fopen("welcome.txt","r") or exit("Khong mo duoc file");
```

```
if(feof($file)){
```

```
    echo "Ket thuc file"; fclose($file);
```

```
}
```

# PHP for Base

## ❖ File in PHP

- **fgets()** là phương thức sử dụng để đọc từng dòng trên file trong PHP
- Cú pháp: **fgets**(\$file\_can\_doc)

Ví dụ:

```
$file=fopen("welcome.txt","r") or exit("Khong mo duoc file");  
while(!feof($file)){  
    echo fgets($file) . "<br/>";  
}; fclose($file);
```

# PHP for Base

## ❖ File in PHP

- **fgetc()** là phương thức sử dụng để đọc từng ký tự trên file trong PHP
- Cú pháp: **fgetc**(\$file\_can\_doc)

Ví dụ:

```
$file=fopen("welcome.txt","r") or exit("Khong mo duoc file");  
  
while(!feof($file)){  
    echo fgetc($file) . "<br/>";  
}; fclose($file);
```

# PHP for Base

## ❖ File in PHP

- Một số phương thức khác làm việc với file trong PHP

Method	Description
<b>basename()</b>	<p>Trả về tên file từ một đường dẫn Cú pháp: <code>basename(path,suffix)</code> suffix: Sử dụng nếu có để loại bỏ phần thông tin trong file ví dụ như định dạng: <code>\$path = "/stanfordcorp/home.php";</code> <code>echo basename(\$path) . "&lt;br/&gt;";</code> <code>echo basename(\$path, ".php");</code></p>
<b>copy()</b>	<p>Sử dụng để copy file Cú pháp: <code>copy(file,to_file)</code> Ví dụ: <code>echo copy("source.txt","target.txt");</code></p>

# PHP for Base

## ❖ File in PHP

- Một số phương thức khác làm việc với file trong PHP

Method	Description
<b>fread()</b>	Đọc dữ liệu từ một file Cú pháp: fread(file,length) Ví dụ: \$file = fopen("test.txt","r"); fread(\$file,filesize("test.txt")); fclose(\$file);
<b>fwrite()</b>	Ghi dữ liệu vào file mở Cú pháp: fwrite(file,string,length) Ví dụ: \$file = fopen("test.txt","w"); echo fwrite(\$file,"Hello World. Testing!"); fclose(\$file);

# PHP for Base

## ❖ File in PHP

- Một số phương thức khác làm việc với file trong PHP

Method	Description
<b>dirname()</b>	Trả về tên thư mục trong đường dẫn Ví dụ: <code>echo dirname("c:/stanfordcorp/home.php");</code> Out: <code>c:/stanfordcorp</code>
<b>disk_free_space()</b>	Trả về dung lượng trống dạng byte của thư mục Cú pháp: <b>disk_free_space(thu_muc)</b>
<b>disk_total_space()</b>	Tổng dung lượng của thư mục Cú pháp: <b>disk_total_space(thu_muc)</b>
<b>diskfreespace()</b>	Là một alias giống như hàm <b>disk_free_space()</b>



# PHP for Base

## ❖ File in PHP

- Một số phương thức khác làm việc với file trong PHP

Method	Description
<b>fgetss()</b>	Trả về từng dòng dữ liệu trong đó sẽ loại bỏ các thẻ HTML, PHP trong file Cú pháp: <code>fgetss(file,length,tags)</code>
<b>file()</b>	Trả về mảng với từng dòng sẽ được gán vào từng phần tử của mảng. Cú pháp: <code>print_r(file("test.txt"));</code>
<b>file_exists()</b>	Kiểm tra file có tồn tại hay không Cú pháp: <code>file_exists(path)</code>

# PHP for Base

## ❖ File in PHP

- Một số phương thức khác làm việc với file trong PHP

Method	Description
<b>file_get_contents</b>	<p>Trả về nội dung của một file về dạng chuỗi</p> <p>Cú pháp:</p> <p><b>file_get_contents</b>(path,include_path,context,start,max_length)</p> <p>Trong đó:</p> <ul style="list-style-type: none"><li>- <b>path</b>: bắt buộc phải có</li><li>- <b>include_path</b>: thiết lập tham số bằng 1 nếu muốn tìm file bao gồm cả *.ini</li><li>- <b>context</b>: xác định nội dung của file xử lý. Có thể bỏ qua nếu NULL</li><li>- <b>start, max_length</b>: Là vị trí bắt đầu cần lấy, độ lớn tối đa cần lấy.</li></ul>

# PHP for Base

## ❖ File in PHP

- Một số phương thức khác làm việc với file trong PHP

Method	Description
<b>filetime()</b>	Trả về thời gian truy xuất lần cuối của file Cú pháp: filetime(filename)
<b>filectime()</b>	Trả về thời gian cuối thay đổi file Ví dụ: echo "Last change: ".date("F d Y H:i:s.",filectime("test.txt"));
<b>filesize()</b>	Hàm trả về kích thước của một file Cú pháp: filesize(filename)
<b>filetype()</b>	Hàm trả về kiểu file xác định bởi một file hoặc thư mục, bao gồm: fifo, char, dir, block, link, file, unknown
<b>is_dir()</b>	Hàm kiểm tra có phải là thư mục hay không

# PHP for Base

## ❖ File in PHP

- Một số phương thức khác làm việc với file trong PHP

Method	Description
<b>is_executable</b>	Kiểm tra xem có phải là file thực thi
<b>is_file</b>	Kiểm tra xác định file hay sử dụng
<b>is_link()</b>	Kiểm tra file xác định là một link
<b>pathinfo()</b>	<p>Hàm trả về một mảng chứa thông tin path</p> <p>Cú pháp: pathinfo(path,options)</p> <p>Trong đó:</p> <ul style="list-style-type: none"><li>- <b>path</b>: yêu cầu cần có</li><li>- <b>Options</b>: có thể có với<ul style="list-style-type: none"><li>+ PATHINFO_DIRNAME - return only dirname</li><li>+ PATHINFO_BASENAME - return only basename</li><li>+ PATHINFO_EXTENSION - return only extension</li></ul></li></ul>

# PHP for Base

## ❖ File in PHP

- Một số phương thức khác làm việc với file trong PHP

Method	Description
<b>rename()</b>	Phương thức thay đổi tên một file hoặc thư mục
<b>rmdir()</b>	Hàm remove một thư mục rỗng trong PHP
<b>tempnam()</b>	Tạo ra một file tạm trong một thư mục xác định Cú pháp: tempnam(dir,prefix) Ví dụ: echo tempnam("C:\wamp\www","TMP");
<b>tmpfile()</b>	Phương thức tạo ra một file tạm có tên duy nhất trong kiểu đọc và ghi Ví dụ: \$temp = tmpfile(); fwrite(\$temp, "Stanford - Day lap trinh"); rewind(\$temp); echo fread(\$temp,1024); fclose(\$temp);

# PHP for Base

## ❖ File in PHP

- Một số phương thức khác làm việc với file trong PHP

Method	Description
<b>unlink()</b>	<p>Thực hiện để xóa file, trả về giá trị TRUE nếu xóa thành công ngược lại thất bại sẽ trả về FALSE.</p> <p>Ví dụ:</p> <pre>&lt;?php \$fh = fopen('test.html', 'a'); fwrite(\$fh, '&lt;h1&gt;Hello world!&lt;/h1&gt;'); fclose(\$fh);  unlink('test.html'); ?&gt;</pre>

# PHP for Base

## File Upload

# PHP for Base

## ❖ File Upload

- Cho phép người dùng tải file lên thư mục được lưu trữ trên server
- Trong đó thẻ form cần có thuộc tính enctype="multipart/form-data"
- Sử dụng biến dạng mảng **\$\_FILES** để làm việc với files trong PHP

```
<?php
if ($_FILES["file"]["error"] > 0)
{
    echo "Error: " . $_FILES["file"]["error"] . "<br>";
}
else
{
    echo "Upload: " . $_FILES["file"]["name"] . "<br>";
    echo "Type: " . $_FILES["file"]["type"] . "<br>";
    echo "Size: " . ($_FILES["file"]["size"] / 1024) . " kB<br>";
    echo "Stored in: " . $_FILES["file"]["tmp_name"];
}
?>
```



# PHP for Base

## ❖ File Upload

### ▪ Các thuộc tính của File Upload:

- `$_FILES["file"]["name"]` : tên file upload
- `$_FILES["file"]["type"]` : kiểu file upload
- `$_FILES["file"]["size"]` : kích thước file upload
- `$_FILES["file"]["tmp_name"]` : tên file tạm được lưu trữ trên server
- `$_FILES["file"]["error"]` : lỗi khi upload file

**Trong đó:** file là **id** của thẻ input trong HTML có type là **file**

# PHP for Base

## ❖ File Upload

- Sử dụng phương thức **move\_uploaded\_file()** để thực hiện di chuyển file từ thư mục temp sang thư mục chính thức trên server.
- **Cú pháp:**

**move\_uploaded\_file(ten\_file\_tam, thu\_muc\_luu\_tru);**

Ví dụ:

```
move_uploaded_file($_FILES["file"]["tmp_name"],  
"upload/" . $_FILES["file"]["name"]);
```

# PHP for Base

## ❖ File Upload

- Ví dụ: Thực hiện upload file trong PHP

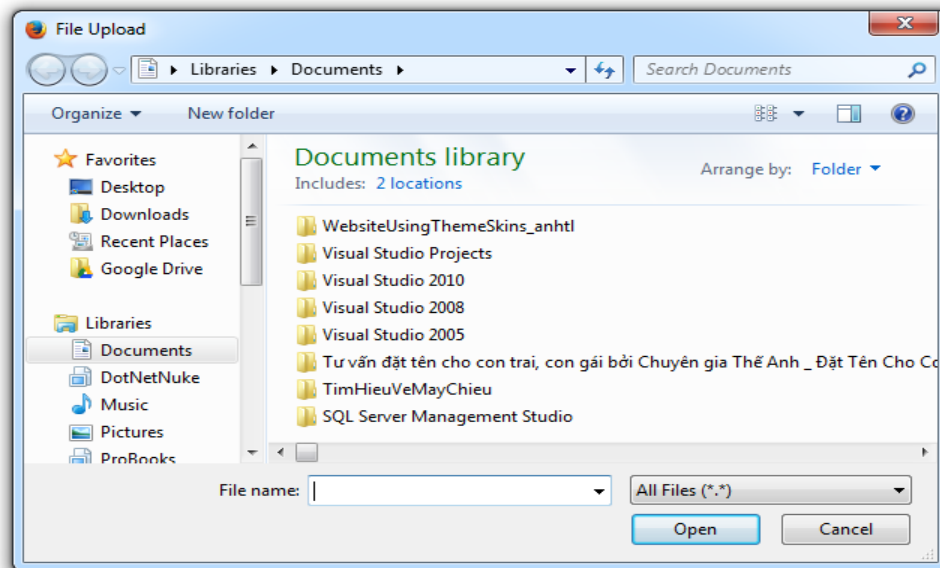
```
<form action="upload_action.php" method="post" enctype="multipart/form-data">
```

```
Chọn file: <input type="file" name="file" id="file"/>
```

```
<input type="submit" name="btnUpload" value="Tải file"/>
</form>
```

Giao diện:

Chọn file:  No file selected.



# PHP for Base

## ❖ File Upload

- Ví dụ: Thực hiện upload file trong PHP

```
<?php

if ($_FILES["file"]["error"] > 0)
{
    echo "Return Code: " . $_FILES["file"]["error"] . "<br>";
}
else
{
    echo "Upload: " . $_FILES["file"]["name"] . "<br>";
    echo "Type: " . $_FILES["file"]["type"] . "<br>";
    echo "Size: " . ($_FILES["file"]["size"] / 1024) . " kB<br>";
    echo "Temp file: " . $_FILES["file"]["tmp_name"] . "<br>";

    if (file_exists("upload/" . $_FILES["file"]["name"]))
    {
        echo $_FILES["file"]["name"] . " already exists. ";
    }
    else
    {
        move_uploaded_file($_FILES["file"]["tmp_name"],
            "upload/" . $_FILES["file"]["name"]);
        echo "Stored in: " . "upload/" . $_FILES["file"]["name"];
    }
}

?>
```



**Thank You !**