

1

PHP Include File

PHP Include Another File

2

- ❑ Insert the content of **one PHP file into another PHP** file before the server executes it. In this way, you have already used the `db_config.php` during the CRUD operations.
- ❑ To do this use:
 - ❑ `include()` generates a **warning**, but the script will **continue execution**
 - ❑ `require()` generates a **fatal error**, and the script will **stop**

include() example

3

```
# menu.php
```

```
<a href="/default.php">Home</a>
<a href="/tutorials.php">Tutorials</a>
<a href="/references.php">References</a>
<a href="/examples.php">Examples</a>
<a href="/contact.php">Contact Us</a>
```

PHP

```
# home.php
```

```
<body>
<div class="leftmenu">
<?php include("menu.php"); ?>
</div>

<h1>Welcome to my home page.</h1>
<p>I have a great menu here.</p>

</body>
```

4

PHP File Input/Output

PHP file I/O functions

5

function name(s)	category
<u>file</u> , <u>file_get_contents</u> , <u>file_put_contents</u>	reading/writing entire files
<u>basename</u> , <u>file_exists</u> , <u>filesize</u> , <u>fileperms</u> , <u>filemtime</u> , <u>is_dir</u> , <u>is_readable</u> , <u>is_writable</u> , <u>disk_free_space</u>	asking for information
<u>copy</u> , <u>rename</u> , <u>unlink</u> , <u>chmod</u> , <u>chgrp</u> , <u>chown</u> , <u>mkdir</u> , <u>rmdir</u>	manipulating files and directories
<u>glob</u> , <u>scandir</u>	reading directories

Reading/writing files

6

contents of foo.txt	file ("foo.txt")	file_get_contents ("foo.txt")
Hello how are you? I'm fine	array("Hello\n", #0 "how are\n", #1 "you?\n", #2 "\n", #3 "I'm fine\n" #4)	"Hello\n how are\n you?\n \n I'm fine\n"

- `file` returns lines of a file as **an array**
- `file_get_contents` returns entire **contents** of a file as a **string**

Reading/writing an entire file

7

```
# reverse a file
$text = file_get_contents("poem.txt");
$text = strrev($text);
file_put_contents("poem.txt", $text);
```

PHP

- `file_get_contents` returns entire contents of a file as a **string**
- `file_put_contents` **writes** a string into a file, **replacing** any prior contents

Appending to a file

8

```
# add a line to a file
$new_text = "P.S. ILY, GTG TTYL!~";
file_put_contents("poem.txt", $new_text,
FILE_APPEND);
```

PHP

old contents

Roses are red,
Violets are blue.
All my base,
Are belong to you.

new contents

Roses are red,
Violets are blue.
All my base,
Are belong to you.
P.S. ILY, GTG TTYL!~

The file function

9

```
# display lines of file as a bulleted list
$lines = file("todolist.txt");
foreach ($lines as $line) {
    ?>
    <li> <?= $line ?> </li>
<?php
}
?>
```

PHP

- file returns the lines of a file as an **array of strings**
 - ▣ each string **ends with \n**
 - ▣ to strip the \n off each line, use optional second parameter:

```
$lines = file("todolist.txt", FILE_IGNORE_NEW_LINES);
```

PHP

Unpacking an array: `list`

10

```
list($var1, ..., $varN) = array;
```

PHP

```
$values = array("mundruid", "18", "f", "96");  
...  
list($username, $age, $gender, $iq) = $values;
```

PHP

- the `list` function accepts a comma-separated list of variable names as parameters
- use this to quickly "**unpack**" an array's contents into several variables

Fixed-length files, file and list

11

```
Xenia Mountrouidou  
(919) 685-2181  
570-86-7326
```

personal.txt

```
list($name, $phone, $ssn) = file("personal.txt");
```

PHP

- reads the file into an array of lines and unpacks the lines into variables
- Need to know a file's exact length/format

Splitting/joining strings

12

```
$array = explode(delimiter, string);  
$string = implode(delimiter, array);
```

PHP

```
$str = "CS 380 01";  
$arr = explode(" ", $str); # ("CS", "380", "01")  
$jstr = implode("...", $arr); # "CSE...380...01"
```

PHP

□ **explode and implode** convert between strings and arrays

Example explode

13

```
Harry Potter, J.K. Rowling  
The Lord of the Rings, J.R.R. Tolkien  
Dune, Frank Herbert
```

contents of input file books.txt

```
<?php foreach (file("books.txt") as $book) {  
    list($title, $author) = explode("", $book);  
    ?>  
    <p> Book title: <?= $title ?>, Author: <?=  
$author ?> </p>  
<?php  
}  
?>
```

PHP

Reading directories

14

function	description
<u>scandir</u>	returns an array of all file names in a given directory (returns just the file names, such as "myfile.txt")
<u>glob</u>	returns an array of all file names that match a given pattern (returns a file path and name , such as "foo/bar/myfile.txt")

Example for glob

15

```
# reverse all poems in the poetry directory
$poems = glob("poetry/poem*.dat");
foreach ($poems as $poemfile) {
    $text = file_get_contents($poemfile);
    file_put_contents($poemfile, strrev($text));
    print "I just reversed " .
basename($poemfile);
}
```

PHP

- ❑ glob can match a "wildcard" path with the * character
- ❑ the **basename** function strips any leading directory from a file path

Example for scandir

16

```
<ul>
<?php
$folder = "taxes/old";
foreach (scandir($folder) as $filename) {
    ?>
    <li> <?= $filename ?> </li>
<?php
}
?>
</ul>
```

PHP

- .
- ..
- 2009_w2.pdf
- 2007_1099.doc

output

17

PHP Exceptions

Exceptions

18

- Used to change the normal flow of the code execution if a specified error (exceptional) condition occurs.
- What normally happens when an exception is triggered:
 - current code state is saved
 - code execution will switch to a predefined (custom) exception handler function
 - the handler may then
 - **resume** the execution from the saved code state,
 - **terminate** the script execution or
 - **continue** the script from a different location in the code

Exception example

19

```
<?php
//create function with an exception
function checkStr($str)
{
    if(strcmp($str, "correct") != 0)
    {
        throw new Exception("String is not correct!");
    }
    return true;
}

//trigger exception
checkStr("wrong");
?>
```

PHP

Exception example (cont.)

20

```
<?php
//create function with an exception
function checkStr($str)
{
    ...
}

//trigger exception in a "try" block
try
{
    checkStr("wrong");
    //If the exception is thrown, this text will not be shown
    echo 'If you see this, the string is correct';
}

//catch exception
catch(Exception $e)
{
    echo 'Message: ' . $e->getMessage();
}
```

Thank You