Chapter 3

File Handling

Contents

- Files and Directories
- Write to Files
- Read from Files
- Create Directories
- Upload Files
- Rename and Delete Files and Directories

PHP File Handling

- File handling is an important part of any web application.

- You often need to open and process a file for different tasks.

- PHP has several functions for creating, reading, and editing files.

- The files can be .doc file, .txt file, .xml file any kind of file supports all php function for manipulate files.

PHP File Open and Read

- how to open file , read file and close file using file handling functions:

    1. fopen() – open file
    2. fread() – read file
    3. fclose() – close file

PHP Create File - fopen()

- The fopen() function is used to create a file..
- If you use fopen() on a file that does not exist, it will create it.
- The example below creates a new file called "testfile.txt".
- The file will be created in the same directory where the PHP code resides:

  ```
  $myfile = fopen("testfile.txt", "w");
  ```

File functions	The use of functions
touch()	used to create a file.
unlink()	used to delete a file.
copy()	used to copy a file.
rename()	used to rename a file.
file_exists()	used to check whether the file exists or not.
filesize()	used to check size of file.
realpath()	used to check real path of file.
fopen()	used to open existing file.
fread()	used to reads from an pen file.
fwrite()	used to write to file.
fclose()	used to close an open file.
fgets()	used to read a single line from a file.
fgetc()	used to read a single character from a file.
feof()	used to check 'end of file'.

PHP fopen() function

- PHP fopen() function used to open a file. If file does not exist then fopen() function will create a new file.
- The fopen() function must use with mode character like 'w', 'a' ,'r' etc.

```php
<?php
fopen("filename with extension", "mode char");
?>
```

Modes	Description
r	**Open a file for read only**. File pointer starts at the beginning of the file
w	**Open a file for write only**. Erases the contents of the file or creates a new file if it doesn't exist. File pointer starts at the beginning of the file
a	**Open a file for write only**. The existing data in file is preserved. File pointer starts at the end of the file. Creates a new file if the file doesn't exist
x	**Creates a new file for write only**. Returns FALSE and an error if file already exists
r+	**Open a file for read/write**. File pointer starts at the beginning of the file
w+	**Open a file for read/write**. Erases the contents of the file or creates a new file if it doesn't exist. File pointer starts at the beginning of the file
a+	**Open a file for read/write**. The existing data in file is preserved. File pointer starts at the end of the file. Creates a new file if the file doesn't exist
x+	**Creates a new file for read/write**. Returns FALSE and an error if file already exists

Example - fopen()

```php
<?php
//open text file
fopen("abc.txt","w");
//open ms word .doc file
fopen("abc.doc","w");
//open pdf file
fopen('abc.pdf',"w");
?>
```

PHP Read File - fread()

- The fread() function reads from an open file.
- The first parameter of fread() contains the name of the file to read from and the second parameter specifies the maximum number of bytes to read.
- fread($myfile,filesize("webdictionary.txt"));

Example

- ```
  <!DOCTYPE html>
  <html>
  <body>

  <?php
  $myfile = fopen("webdictionary.txt", "r") or die("Unable
  to open file!");
  echo fread($myfile,filesize("webdictionary.txt"));
  fclose($myfile);
  ?>

  </body>
  </html>
  ```

PHP close file – fclose()

- The fclose() function is used to close an open file.

- PHP fclose() syntax

<?php

fclose("filename");

?>

PHP File Create and Write

- how to create a file and how to write to a file on the server.
- Create a File – touch() ,
- Create a File, if does not exist – fopen()
- Write to a File – fwrite()

touch() and fopen()

```php
<?php
//create text file
touch("abc.txt");
//create ms word .doc
   file
touch("abc.doc");
//create pdf file
touch('abc.pdf');
?>
```

```php
<?php
//create text file
fopen("abc.txt","w");
//create word .doc file
fopen("abc.doc","w");
//create pdf file
fopen('abc.pdf',"w");
?>
```

PHP Write to File - fwrite()

- The fwrite() function is used to write to a file.

- The first parameter of fwrite() contains the name of the file to write to and the second parameter is the string to be written.

- The example below writes a couple of names into a new file called "newfile.txt":

```php
<?php
$myfile = fopen("newfile.txt", "w") or die("Unable
    to open file!");
$txt = "John Doe\n";
fwrite($myfile, $txt);
$txt = "Jane Doe\n";
fwrite($myfile, $txt);
fclose($myfile);
?>
```

- <?php

```php
//open file abc.txt
$myfile = fopen("abc.txt", "w");
$text = "Meera Academy";
fwrite($myfile, $text);
fclose($myfile);
?>
```

PHP form example of fwrite()

```
<html>
<body>
<FORM method="POST">
Enter String : <input type="text" name="name">
   <br/> <br/>
<input type="submit" name="Submit1"
   value="Write File">
</FORM>
```

Cont...

```php
<?php
if(isset($_POST['Submit1'])) {
 //open file abc.txt in append mode
$myfile = fopen("abc.txt", "a");
$text = $_POST["name"];
fwrite($myfile, $text);
fclose($myfile); }
?> </body> </html>
```

PHP Close File - fclose()

- The fclose() function is used to close an open file.

- It's a good programming practice to close all files after you have finished with them.

- The fclose() requires the name of the file (or a variable that holds the filename) we want to close:

```php
<?php
$myfile = fopen("webdictionary.txt", "r");
// some code to be executed....
fclose($myfile);
?>
```

PHP Check End-Of-File - feof()

- The feof() function checks if the "end-of-file" (EOF) has been reached.
- The example below reads the "webdictionary.txt" file line by line, until end-of-file is reached:

```php
<?php
$file = fopen("abc.txt", "r");
while(! feof($file)) {
  echo fgets($file). "<br>";
}
fclose($file);
?>
```

PHP Read Single Character - fgetc()

- The fgetc() function is used to read a single character from a file.

```php
<?php
  $myfile =
  fopen("webdictionary.txt", "r") or die("Unable to
  open file!");
  // Output one character until end-of-file
  while(!feof($myfile)) {
    echo fgetc($myfile);
  }
  fclose($myfile);
?>
```

PHP copy() Function

- The copy() function copies a file.
- **Note:** If the *to_file* file already exists, it will be overwritten.
- Syntax

copy(*from_file*, *to_file*, *context*)

```php
<?php
Echo copy("webdictionary.doc","studMark.doc");
?>
```

unlink() Function

- Delete a file:

```php
<?php
unlink("stud.doc");
?>
```

PHP file_exists() Function

- The file_exists() function checks whether a file or directory exists.

Syntax

file_exists(*path*)

- *Check whether a file exists:*

```php
<?php
echo file_exists("stud1.txt");
?>
```

filesize() Function

- The filesize() function returns the size of a file.
- Return the file size for "test.txt":

```php
<?php
echo filesize("test.txt");
?>
```

PHP File Inclusion

- PHP has two function which can used to include one PHP file into another PHP file before the server executes it.

    1. The include() function
    2. The require() function

- For the designing purpose in web forms the same header, footer or menu displayed on all web pages of website.

- Programmer has to design same menu on multiple pages, if changes required in future it will very complicated to open all pages then make change on all pages.
- for resolving this problem we use include and require function in php.
- Just design menu or header in one php page and display same menu on multiple pages using include function.
- if changes required on menu.php page, it will make effect on all other pages automatically.

The include() function

- The include function copy all text of one PHP file into another PHP file that used the include statement.

- The include function used when we use same menu or header on multiple pages of a website.

- *PHP include() syntax:*

  include ' filename';

Example1

```php
<?php
echo "<p>Copyright &copy; 1999-" . date("Y") . " Infolink.com</p>";
?>
```

```html
<html>
<body>

<h1>Welcome to my home page!</h1>
<p>Some text.</p>
<p>Some more text.</p>
<?php include 'footer.php';?>

</body>
</html>
```

Example2

Assume we have a standard menu file called "menu.php":

```php
<?php
echo '<a href="/default.asp">Home</a> -
<a href="/html/default.asp">HTML
Tutorial</a> -
<a href="/css/default.asp">CSS Tutorial</a> -
<a href="/js/default.asp">JavaScript
Tutorial</a> -
<a href="default.asp">PHP Tutorial</a>';
?>
```

```html
<html>
<body>

<div class="menu">
<?php include 'menu.php';?>
</div>

<h1>Welcome to my home
page!</h1>
<p>Some text.</p>
<p>Some more text.</p>

</body>
</html>
```

Example3

Assume we have a file called "vars.php", with some variables defined:

```php
<?php
$color='red';
$car='BMW';
?>
```

Then, if we include the "vars.php" file, the variables can be used in the calling file:

```php
<html>
<body>

<h1>Welcome to my home page!</h1>
<?php include 'vars.php';
echo "I have a $color $car.";
?>

</body>
</html>
```

PHP include Example

- We have a same header for all website pages so create "header.php" file like:

```php
<?php
echo "<h1> Welcomme
    Meera Academy
    </h1>";
?>
```

```
<html>
 <body>
<?php include 'header.php'; ?>
<p>The first page of site</p>
    </body>
</html>
```

Welcome to Meera Academy

The first page of site

Example-2

Menu.php

```php
<?php echo '<ul><li><a
   href="home.com">HOME</a></li> <li><a
   href="php.com">PHP</a></li> <li><a
   href="asp.com">ASP.NET</a></li> <li><a
   href="project.com">PROJECTS</a></li></ul>';
   ?>
```

```html
<html>
 <head>
<title>PHP include Example</title>
</head>
 <body>
<?php include 'menu.php'; ?>
<p>The first page of site</p>
</body>
</html>
```

The PHP require() Function

- The require() function copy all the text from one php file into another php file that uses the require() function.

- In require() function there is a problem with file then the require() function generate fatal error and stop execution of code.

- while the include() function will continue to execute script.

- The require() function is better than the include() function, because scripts not to be continue if file has problem or missing.