ITMD 462/562
Web Site Application Development

Lecture 5

Fall 2015 – September 23, 2015

Tonight’s Agenda

- Uploading files with POST
- Review AS1 Possible Solution
- Working with the file system
- Saving/reading PHP data to/from a file

Uploading Files

POST File Uploads

- Tutorial examples on how to do file uploads with a POST form
- http://php.net/manual/en/features.file-upload.post-method.php
- http://www.sitepoint.com/file-uploads-with-php/
- Let’s look at the php.net example.

POST File Uploads

- Important things to note.
- Form tag must have the enctype attribute with the value being “multipart/form-data” or it will not work
- MAX_FILE_SIZE hidden field can be used to save the user the trouble up uploading a file bigger than php is configured for but can be tricked.
- File is uploaded to a temporary directory on the server
- All file metadata is stored in the PHP superglobal $_FILES[‘fieldname’] array
- Use the move_uploaded_file function to move the file from the temp directory to your final location
- http://php.net/manual/en/features.file-upload.post-method.php

Assignment 1

Possible approach to Assignment 1

- Demo

PHP File System Use

Reading, Writing, Moving, and any other file needs

File System Related Extensions

- These functions and objects are used to interact with the underlying file system on the server
- http://php.net/manual/en/refs.fileprocess.file.php
- We will concentrate on the Filesystem extension tonight
- Often we need to read the contents of a file to process the data
- Often we want to save the state of our application into a file for later use
- Sometimes we just want to write lines to a file for logging or other purposes

Opening a file

- First need to open a file which returns a resource handle. fopen function

```
<?php
   $handle = fopen("data.txt", "r");
?>
```

- First parameter is the file path, second is the file mode.
- http://php.net/manual/en/function.fopen.php

```
<?php
   if(file_exists("data.txt")){
      $file = fopen("data.txt", "r");
   } else{
      die("Error: The file you are trying to access doesn't exist.");
   }
?>
```

Closing a file

- When finished working with the file it needs to get closed. fclose function

```
<?php
   $file = "data.txt”;

   // Open the file for reading
   $handle = fopen($file, "r") or die("ERROR: Cannot open the file");

   // Some code to be executed

   // Closing the file handle
   fclose($handle);
?>
```

- http://php.net/manual/en/function.fclose.php

Reading Strings of Characters

- Many PHP functions for reading in data from a file. 1 character to whole file.
- The `fread` function can read strings of characters.

```
fread(file handle, length in bytes)
```

- http://php.net/manual/en/function.fread.php

Reading a String of fixed length

```
<?php
$file = "data.txt";

// Open the file for reading
$handle = fopen($file, "r") or die("ERROR: Cannot open the file");

// Read in the entire file
$content = fread($handle,"20");

// Closing the file handle
fclose($handle);

// Display the file content
echo $content;
?>
```

Reading an Entire File

- The fread function can be used with the filesize function to read the whole file at one time.
- Use the filesize function in place of the length in bytes parameter of fread

```
$content = fread($handle, filesize($file));
```

Easier way to read entire file

- You can use the file_get_contents function to read the entire file into a string.
- http://php.net/manual/en/function.file-get-contents.php

```
<?php
$file = "data.txt";

// Reading the entire file into a string
$content = file_get_contents($file) or die("ERROR: Cannot
open the file");

// Display the file content
echo $content;
?>
```

Yet another way

- You can use the php file function
- Similar to file_get_contents but it reads the contents into an array of lines vs a single string
- http://php.net/manual/en/function.file.php

```
<?php
$file = "data.txt";

// Reading the entire file into an array
$arr = file($file) or die("ERROR: Cannot open the file");
   foreach($arr as $line){
   echo $line;
}
?>
```

Basic file writing

- Many PHP functions for writing data to a file.
- The fwrite function can write strings to a file.
- Make sure fopen mode is writable

```
fwrite(file handle, string data)
```

- http://php.net/manual/en/function.fwrite.php

Basic writing example

```
 <?php
$file = "note.txt";

// String of data to be written
$data = "The quick brown fox jumps over the lazy dog.";

// Open the file for writing
$handle = fopen($file, "w") or die("ERROR: Cannot open the file");

// Write data to the file
fwrite($handle, $data) or die ("ERROR: Cannot write the file");

// Closing the file handle
fclose($handle);

echo "Data written to the file successfully";
?>
```

Alternative way

- You can use the file_put_contents to write data to file without opening it
- Can use the FILE_APPEND parameter to append vs overwrite
- http://php.net/manual/en/function.file-put-contents.php

```
<?php
$file = "note.txt";

// String of data to be written
$data = "The quick brown fox jumps over the lazy dog.";

// Write data to the file
file_put_contents($file, $data) or die("ERROR: Cannot write the
file");

echo "Data written to the file successfully";
?>
```

Many other filesystem function

- http://php.net/manual/en/ref.filesystem.php

fread()	Reads a string of characters from a file.
fwrite()	Writes a string of characters to a file.
fgetc()	Reads a single character at a time.
feof()	Checks to see if the end of the file has been reached.
fgets()	Reads a single line at a time.
fgetcsv()	Reads a line of comma - separated values.
fputcsv()	Format line as CSV and write to file pointer
file()	Reads an entire file into an array.
file_get_contents()	Reads an entire file into a string without needing to open it.
file_put_contents()	Writes a whole string to a file without needing to open it.
fpassthru()	Displays the contents of an open file.
readfile()	Displays the contents of a file without needing to open it.
fseek()	Moves the file pointer to a specific location within an open file.
ftell()	Returns the position of the file pointer.
rewind()	Moves the file pointer to the start of the file.

Reading and Writing Data in CSV

- Comma Separated Values (CSV) is a common data format.
- Values are stored in a text file, each row is one item, each items columns are separated by comma.
- Might have a header row or not
- PHP has a couple functions that help us with this task
- `fgetcsv` and `fputcsv`
- http://php.net/manual/en/function.fgetcsv.php
- http://php.net/manual/en/function.fputcsv.php

Object Serialization

- Serialization is a way to represent an Object as a string of text for storage in a file or database.
- Two primary functions `serialize` and `unserialize`
- Serialize creates the data string from the object and unserialize recreates the object from the string.
- http://php.net/manual/en/language.oop5.serialization.php
- http://php.net/manual/en/function.serialize.php
- http://php.net/manual/en/function.unserialize.php

Assignments

Assignments

- Assignment 2 will be finalized and released this weekend.