PHP and File I/O

CS/IT 490 WD, Fall 2013

Last update 2013-09-19

Written by Rachel J. Morris

Breakdown

- JSON Format
- CSV Format
- Handling a file
- Saving a file
- Loading a file

Written by Rachel J. Morris

JSON Format

- JSON – JavaScript Object Notation is a data format that is essentially key-value pairs and arrays.
- http://www.json.org/
- Looks like this:

Key/Value pairs { “name” : “guybrush”, “age” : 25 }

Ordered Array: [9.99, 3.99, 2.99]

Written by Rachel J. Morris

JSON Format

- Like with PHP associative arrays, you can store JSON objects within JSON objects within JSON objects.
- PHP can also easily parse JSON format files into arrays, with only a function call.
- PHP can also convert an array to a JSON string, with only a function call.

CSV Format

- CSV (Comma-Separated Value) is a file format that allows you to create spreadsheets, but in a simple text format.
- Columns are separated by commas , (or some other delimiter), while rows are separated by new-lines.
- PHP can also load in CSV files as an array

CSV Format

```
FIRST NAME,LAST NAME,PAY
Lee,Everett,$12.93
Shawn,Greene,$8.99
Brenda,St. John,$9.99
```


Written by Rachel J. Morris

Handling a file

- You can open a file for reading or writing with **fopen**.
- You specify a flag to tell PHP whether this is a **read**, **write**, or **append**.
- You use **fread** to read the file
- You use **filesize** to get the size of the file
- You use **fclose** to close the file once you're done.

Saving a file

Let's save just a normal text file.

```
<!DOCTYPE html>
<html>
<head>
    <title>PHP Sample</title>
</head>

<body>
    <?
    /* Open a file and write out plaintext */
    $filepath = "files/output-plaintext.txt";

    $fileHandler = fopen( $filepath, "w" ); // Open for write
    if ( $fileHandler == false )
    {
        print_r( error_get_last() );    // Error message (PHP5)
        exit(); // Stop script
    }

    fwrite( $fileHandler, "This is plaintext \n Hello world" );

    fclose( $fileHandler );
    ?>

</body>
</html>
```

Saving a file

If we want to store a JSON or CSV file, first make an array:

```php
$content = array();
$content["Employee1"] = array( "first" => "guybrush", "last" => "threepwood", "occupation" => "pirate" );
$content["Employee2"] = array( "first" => "elaine", "last" => "marley", "occupation" => "mayor" );
$content["Employee3"] = array( "first" => "wally", "last" => "feed", "occupation" => "cartographer" );
```

The same File Open functions will be used, but we will write out the array contents differently...

Saving a file

Saving a JSON file:

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>PHP Sample</title>
5  </head>
6
7  <body>
8      <?
9      $content = array();
10     $content["Employee1"] = array( "first" => "guybrush", "last" => "threepwood", "occupation" => "pirate" );
11     $content["Employee2"] = array( "first" => "elaine", "last" => "marley", "occupation" => "mayor" );
12     $content["Employee3"] = array( "first" => "wally", "last" => "feed", "occupation" => "cartographer" );
13
14     /* Open a file and write out JSON */
15     $filepath = "files/output-json.json";
16
17     $fileHandler = fopen( $filepath, "w" ); // Open for write
18     if ( $fileHandler == false ) { print_r( error_get_last() ); exit(); }
19
20     fwrite( $fileHandler, json_encode( $content ) );
21
22     fclose( $fileHandler );
23     ?>
24
25     <p>Wrote file <?=$filepath?></p>
26
27 </body>
28 </html>
```


Written by Rachel J. Morris

Saving a file

Saving a CSV file (We must iterate through all rows!)

```php
<!DOCTYPE html>
<html>
<head>
    <title>PHP Sample</title>
</head>

<body>
    <?
    $content = array();
    $content["Employee1"] = array( "first" => "guybrush", "last" => "threepwood", "occupation" => "pirate" );
    $content["Employee2"] = array( "first" => "elaine", "last" => "marley", "occupation" => "mayor" );
    $content["Employee3"] = array( "first" => "wally", "last" => "feed", "occupation" => "cartographer" );

    /* Open a file and write out CSV */
    $filepath = "files/output-csv.csv";

    $fileHandler = fopen( $filepath, "w" ); // Open for write
    if ( $fileHandler == false ) { print_r( error_get_last() ); exit(); }

    foreach( $content as $row )
    {
        fputcsv( $fileHandler, $row );
    }

    fclose( $fileHandler );
    ?>

    <p>Wrote file <?=$filepath?></p>

</body>
</html>
```

Loading a file

Reading a plaintext file

```
<?
$filepath = "files/plaintext.txt";

$fileHandler = fopen( $filepath, "r" );
if ( $fileHandler == false )
{
    print_r( error_get_last() );    // Error message (PHP5)
    exit(); // Stop script
}

$contents = fgets( $fileHandler );

fclose( $fileHandler );
?>

<p>File contents:</p>

<p><?=$contents?></p>
```

Loading a file

Reading a JSON file

```php
<?
$filepath = "files/text.json";

$fileHandler = fopen( $filepath, "r" );
if ( $fileHandler == false )
{
    print_r( error_get_last() );    // Error message (PHP5)
    exit(); // Stop script
}

$contents = fgets( $fileHandler );
$jsonContents = json_decode( $contents, false ); // add this

fclose( $fileHandler );
?>

<p>File contents:</p>

<pre>
    <? print_r( $jsonContents ); ?>
</pre>
```

Handy json_decode

Loading a file

Reading a CSV file

```php
<?
$filepath = "files/spreadsheet.csv";

$fileHandler = fopen( $filepath, "r" );
if ( $fileHandler == false )
{
    print_r( error_get_last() );    // Error message (PHP5)
    exit(); // Stop script
}

$rows = array();

while ( $r = fgetcsv( $fileHandler ) )
{
    array_push( $rows, $r );
}

fclose( $fileHandler );
?>

<p>File contents:</p>

<?
foreach( $rows as $row ) {
    echo( "<p>" );
    print_r( $row );
    echo( "</p>" );
}
?>
```

Read one row at a time, while there are still rows to read!

Display each row, one row at a time!

Why File I/O?

- Storing things that don't need a relational database.
- Maybe relatively static content on a webpage
- Easy and fast to read/write
- Lightweight

Why File I/O?

- What would be **good** for storing as simple text, JSON, or CSV?
- What would be **bad** for storing as text, JSON, or CSV instead of in a database?
  - Think of relationships!

References

-

Written by Rachel J. Morris