

CMPT 210

Applications in Software

Lesson 2

Objectives:

- What is PHP and How is it Parsed
- What can PHP Do?
- PHP Start and End Tags
- HTML and PHP
- The Terminator and Errors
- Escape Characters in PHP
- Commenting Your Code

What is PHP:

PHP (Hypertext Preprocess) a server side scripting language. So what does it do? When your web browser accesses a URL such as, <http://localhost/myfirstscript.php>, it makes a request to Apache in this case. The web server wakes up the PHP parsing engine telling it you have to do something before this page is sent back. Then the PHP parsing engine runs through myfirstscript.php and returns the output it finds. This output is passed back to the web server in the form of HTML and then is passed to your browser, in which you see the results.

What can PHP Do?

PHP can do most anything you can think up! PHP was created to create dynamic web pages quickly.

Some of the common uses for PHP and many of them we will learn in this class:

- System functions: Create, open, read, write and close files.
- Gather data from forms: Save the data to a file, send data via e-mail, and return manipulated data back to a user.
- Database manipulations: Create, edit, add, delete and modify databases in a web format.
- Set and access cookies. Yum, what are cookies?
- Start and use session variables.
- Create images on the-fly
- Encrypt data

And these are just a few basic everyday uses of PHP. Your imagination is the limit.

Start and End Tags

```
<?php
```

```
echo "<P>This is a test using the first tag type.</P>";  
?>
```

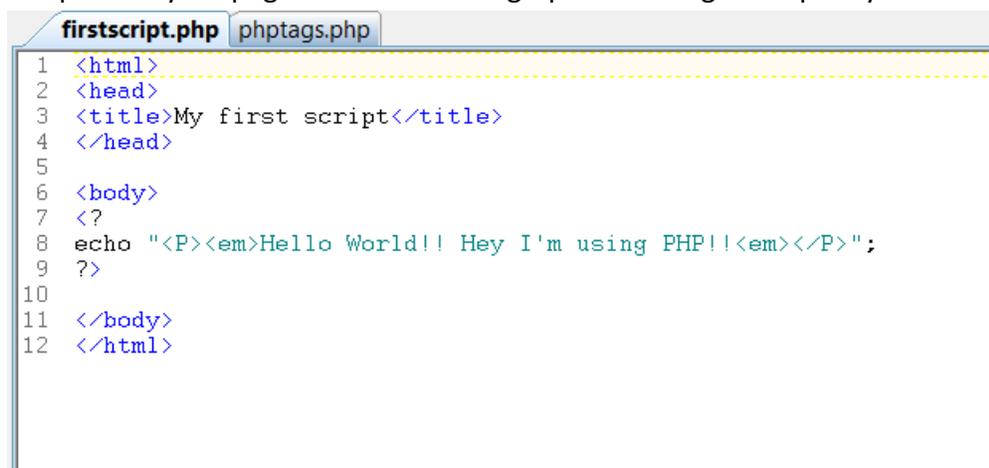
```
<?  
echo "<P>This is a test using the second tag type.</P>";  
?>
```

```
<script language="php">  
echo "<P>This is a test using the third tag type.</P>";  
</script>
```

<http://localhost/lesson1/phptags.php>

PHP and HTML

In the firstscript.php that you used to test XAMPP, there is the use of HTML and PHP in the same script. Most of you are familiar with HTML code and what you can do within it. HTML is what gives the web pages format. You can use HTML to create tables and other layout formats and provide your pages with color and graphics making them pretty.

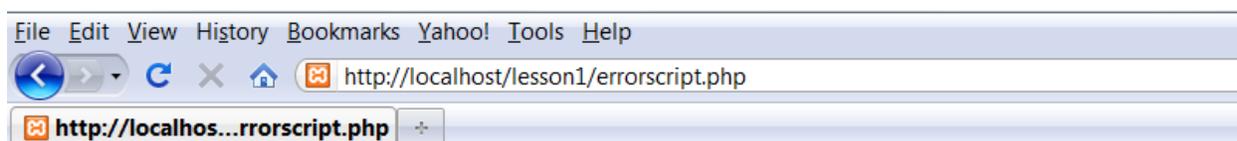


```
firstscript.php  phptags.php  
1 <html>  
2 <head>  
3 <title>My first script</title>  
4 </head>  
5  
6 <body>  
7 <?  
8 echo "<P><em>Hello World!! Hey I'm using PHP!!</em></P>";  
9 ?>  
10  
11 </body>  
12 </html>
```

<http://localhost/lesson1/firstscript.php>

The Terminator and Errors

The terminator also known as the semicolon (;) is required at the end of each command. The terminator instructs the PHP parser that it is done with that line and continue to the next line of code. If you do not terminate a line of code you will come up with the following error.



If you read this error message it tells you what it is looking for and approximately where. It says it is expecting a “,” or “;” and it is on line 8. If you look at the code below it is really line 7 that is missing the terminator. So always look at the line above or sometimes below.

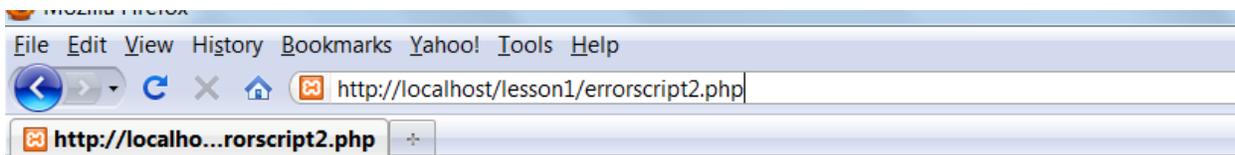
```
errorsript2-a.php | errorsript.php | firstscript.php | phptags.php
1 <HTML>
2 <HEAD>
3 <TITLE>Trying For Another Error</TITLE>
4 </HEAD>
5 <BODY>
6 <?
7 echo "<P>I am trying to produce an error</P>"
8 echo "<P>Was I successful?</p>";
9 ?>
10 </BODY>
11 </HTML>
12
```

<http://localhost/lesson1/errorsript.php>

Escaping your Code

So what the heck does that mean! Well, when you want to use quotation marks within the quotation mark, you need to escape them so the parser will see them as a literal. The following code will produce the error below the code.

```
errorsript2-a.php | errorsript.php | firstscript.php | phptags.php
1 <HTML>
2 <HEAD>
3 <TITLE>Trying For Another Error</TITLE>
4 </HEAD>
5 <BODY>
6
7
8 <?
9 echo "<P>I think this is really "cool"!</p>";
10 ?>
11
12 </BODY>
13 </HTML>
14
```



Parse error: syntax error, unexpected T_STRING, expecting ',' or ';' in C:\XAMPP\htdocs\lesson1\errorsript2.php on line 9

<http://localhost/lesson1/errorsript2.php>

This error message is not as descriptive or helpful as before, but we know that it is on or about line 9. Note the colors in TextPad! Hmm... Doesn't look quite right. So how do we fix this? With the escape code of the backslash “\” before each literal quotation mark. The code will look like the follow fixed version.

```

1 <HTML>
2 <HEAD>
3 <TITLE>Trying For Another Error</TITLE>
4 </HEAD>
5 <BODY>
6
7
8 <?
9 echo "<P>I think this is really \"cool\"!</p>";
10 ?>
11
12 </BODY>
13 </HTML>
14 |

```

<http://localhost/lesson1/errorscript2fixed.php>

Commenting Your Code

Commenting your code is a good habit! I'm going to make it mandatory for this class, meaning I will take points off if code does not have comments. It is very difficult to work on someone else's code without having any comments of what is intent of the code or what does a section do. In most larger companies and small ones too, there has many people working on the same pages of code. It is helpful to know who and what was changed last. This script show the many different ways to comment your code.

```

1 <HTML>
2 <HEAD>
3 <TITLE>Code Comments</TITLE>
4 </HEAD>
5 <BODY>
6
7 <!-- This is an HTML comment. -->
8
9 <?
10
11 // This is a simple PHP comment.
12
13 /* This is a C-style, multi-line comment. You can make this as
14 long as you'd like. */
15
16 # Used to shells? Use this kind of comment. A shell is a piece of software that provides an interface
17 #for users of an operating system which provides access to the services of a kernel.
18
19 ?>
20
21 </BODY>
22 </HTML>
23 |

```

What will this script show and why?

<http://localhost/lesson1/comments.php>

Lab 2

Create the following scripts and have me sign off on the

- **PHP Start and End Tags – phptags.php**
- **HTML and PHP – firstscript.php**
- **The Terminator and Errors – errorscript.php & errorscriptfixed.php**
- **Escape Characters in PHP – errorscript2.php & errorscript2fixed.php**
- **Commenting Your Code – comments.php**