

# Web Programming/Scripting: PHP

*Dr. Michele Weigle*  
Department of Computer Science  
Old Dominion University  
*mweigle@cs.odu.edu*

<http://www.cs.odu.edu/~mweigle/CS312-F09>

1

## What is PHP?

---



- ◆ PHP: Hypertext Preprocessor
- ◆ Server-side scripting language
  - » different from Javascript (a *client-side* scripting language)
- ◆ Free alternative to Microsoft's Active Server Pages (ASP)
- ◆ Can be directly embedded into HTML
- ◆ Syntax is similar to Perl and C
- ◆ PHP is often combined with a MySQL database (which we won't cover)



- ◆ Syntax
- ◆ Arrays
- ◆ Strings
- ◆ Loops
- ◆ Conditionals
- ◆ Functions
- ◆ Forms
- ◆ PHP and XML

3

## Basic Syntax

---

- ◆ Start with **<?php**
- ◆ End with **?>**
- ◆ Each code line must end with a semicolon
- ◆ Comments
  - » one-line `//`
  - » multi-line `/* ... */`
- ◆ To output text, use **echo ' '**
- ◆ All variables start with **\$**
  - » data type does not need to be set before declaring variable

```
$text = "Hello World";  
$num = 16;
```

4

## Hello World Example

---

```
<html>
  <head><title>PHP Test</title></head>
  <body>
    <?php echo '<p>Hello World</p>'; ?>
  </body>
</html>
```

*file extension must be php*

<http://www.cs.odu.edu/~mweigle/cs312/php/hello.php>

<http://us.php.net/tut.php>

5

## PHPInfo Example

---

- ◆ `phpinfo()` is a built-in function that displays useful information about your system setup (loaded PHP modules, predefined variables, configuration settings)

```
<html>
  <head><title>PHP Info</title></head>
  <body>
    <?php phpinfo(); ?>
  </body>
</html>
```

<http://www.cs.odu.edu/~mweigle/cs312/php/info.php>

<http://us.php.net/tut.php>

6

## Displaying Browser Type

---

- ◆ Let's check what browser the user is using
  - » look at the "User-Agent:" option that is sent in the HTTP request header
  - » \$\_SERVER is a special reserved variable

```
<html>
<head><title>PHP Browser Check</title></head>
<body>
  <?php
    echo $_SERVER['HTTP_USER_AGENT'];
  ?>
</body>
</html>
```

<http://www.cs.odu.edu/~mweigle/cs312/php/browser.php>

<http://us2.php.net/manual/en/reserved.variables.server.php>

7

## Strings in PHP

---

- ◆ Concatenation operator .

```
$txt1 = "Hello World";
$txt2 = "1234";
echo $txt1 . " " . $txt2;
```

Hello World 1234

[http://www.w3schools.com/PHP/php\\_string.asp](http://www.w3schools.com/PHP/php_string.asp)

8

## Useful String Functions

---

- ◆ `strlen(String)`
  - » returns the *length* of the string
  - » number of characters, including spaces
  
- ◆ `strpos(String, SearchString)`
  - » returns the starting position of *SearchString* if found inside *String*
    - ❖ position starts from 0 (i.e., 0 is the first character)
  - » returns FALSE if *SearchString* is not found in *String*

[http://www.w3schools.com/PHP/php\\_string.asp](http://www.w3schools.com/PHP/php_string.asp)

9

## More PHP Syntax

---

- ◆ Familiar C++/Java/Perl operators
  - » comparison: `==`, `!=`, `>`, `>=`, `<`, `<=`
  - » assignment: `=`, `+=`, `*=`, ...
  - » logical: `&&`, `||`, `!`
  
- ◆ Familiar C++/Java/Perl conditionals syntax

```
if (condition) {
    statements;
} elseif {
    statements;
} else {
    statements;
}
```

[http://www.w3schools.com/PHP/php\\_if\\_else.asp](http://www.w3schools.com/PHP/php_if_else.asp)

10

# Displaying Browser Type

## Now With Conditionals

---

*<html>, <head>,  
<body> tags removed  
in remaining examples  
to save space*

```
<?php
if (strpos($_SERVER['HTTP_USER_AGENT'], 'MSIE') != FALSE) {
    echo "You are using Internet Explorer.";
} else {
    echo "You aren't using Internet Explorer, you're using <br />" .
    $_SERVER['HTTP_USER_AGENT'];
}
?>
```

<http://www.cs.odu.edu/~mweigle/cs312/php/browser2.php>

<http://us.php.net/tut.php>

11

# Mixing PHP and HTML

---

- ◆ Logical flow of the script remains intact even if broken up with HTML statements.

```
<?php
if (strpos($_SERVER['HTTP_USER_AGENT'], 'MSIE') != FALSE) {
?>
<h3>strpos() returned non-false</h3>
<p>You are using Internet Explorer</p>
<?php
} else {
?>
<h3>strpos() returned false</h3>
<p>You are not using Internet Explorer, you're using</p>
<?php
    echo $_SERVER['HTTP_USER_AGENT'];
}
?>
```

<http://us.php.net/tut.php>

<http://www.cs.odu.edu/~mweigle/cs312/php/html-php.php>

12

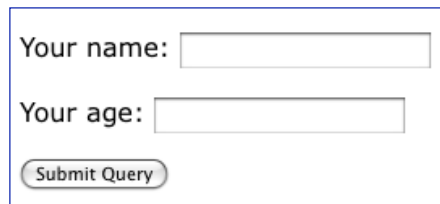
## PHP and Forms

---

- ◆ Use a separate PHP file (not embedded in HTML)

```
<form action="action.php" method="post">
  <p>Your name: <input type="text" name="name" /> </p>
  <p>Your age: <input type="text" name="age" /> </p>
  <p><input type="submit" /> </p>
</form>
```

<http://www.cs.odu.edu/~mweigle/cs312/php/php-form.html>



<http://us.php.net/tut.php>

13

## PHP and Forms

---

```
Hi <?php echo htmlspecialchars($_POST['name']); ?>.
You are <?php echo (int)$_POST['age']; ?> years old.
```

<http://www.cs.odu.edu/~mweigle/cs312/php/action.php>

`htmlspecialchars()` ensures that any special HTML characters are properly encoded so people can't inject HTML tags or JavaScript into your page.

- ◆ `$_POST`
  - » when "post" method is used
- ◆ `$_GET`
  - » when "get" method is used
- ◆ `$_REQUEST`
  - ◆ when either "get" or "post" is used

[http://www.w3schools.com/PHP/php\\_get.asp](http://www.w3schools.com/PHP/php_get.asp)

14

# PHP Arrays

---

## ◆ Three types

### » numeric

❖ array with numeric ID key

### » associative

❖ array where each ID key is associated with a value

### » multidimensional

❖ array containing one or more arrays

[http://www.w3schools.com/PHP/php\\_arrays.asp](http://www.w3schools.com/PHP/php_arrays.asp)

15

# PHP Numeric Arrays

---

## ◆ Creation

```
$names = array("Peter", "Quagmire", "Joe");
```

**OR**

```
$names[0] = "Peter";
```

```
$names[1] = "Quagmire";
```

```
$names[2] = "Joe";
```

Length of an array:  
`count (array)`

`count ($names)`

3

## ◆ Usage

```
$names[1]      Quagmire
```

[http://www.w3schools.com/PHP/php\\_arrays.asp](http://www.w3schools.com/PHP/php_arrays.asp)

16



## PHP Associative Arrays

---

### ◆ Creation

```
$ages = array("Peter"=>"32", "Quagmire"=>"30", "Joe"=>"34");
```

**OR**

```
$ages['Peter'] = "32";
```

```
$ages['Quagmire'] = "30";
```

```
$ages['Joe'] = "34";
```

### ◆ Usage

```
$ages['Peter']           32
```

[http://www.w3schools.com/PHP/php\\_arrays.asp](http://www.w3schools.com/PHP/php_arrays.asp)

17

## PHP Multidimensional Arrays

---

### ◆ Creation

```
$families = array (  
    "Griffin"=>array ("Peter", "Lois", "Megan"),  
    "Quagmire"=>array ("Glenn"),  
    "Brown"=>array ("Cleveland", "Loretta", "Junior")  
);
```

### ◆ Usage

```
$families['Griffin'][2]           Megan
```

[http://www.w3schools.com/PHP/php\\_arrays.asp](http://www.w3schools.com/PHP/php_arrays.asp)

18

# PHP Loops

---

## ◆ Familiar C++/Java syntax

```
while (condition) {  
    statements;  
}  
  
do {  
    statements;  
} while (condition);  
  
for (init; condition; increment)  
{  
    statements;  
}
```

## ◆ Includes Perl-like *foreach* statement

```
foreach (array as item) {  
    statements;  
}
```

```
$arr = array ("one", "two");  
foreach ($arr as $item) {  
    echo $item . "<br />";  
}
```

» Perl syntax is slightly different

[http://www.w3schools.com/PHP/php\\_looping.asp](http://www.w3schools.com/PHP/php_looping.asp)

19

# PHP Functions

---

## ◆ There are over 700 built-in functions available

» <http://www.w3schools.com/PHP/default.asp>

## ◆ Writing your own functions

» begin with the word **function**

❖ syntax similar to Javascript

```
function functionName () {  
    statements;  
}
```

[http://www.w3schools.com/PHP/php\\_functions.asp](http://www.w3schools.com/PHP/php_functions.asp)

20

# PHP Functions

---

## ◆ Functions with parameters

```
function functionName (parameters) {  
    statements;  
}
```

## ◆ Functions with return values

```
function functionName (parameters) {  
    statements;  
    return value;  
}
```

[http://www.w3schools.com/PHP/php\\_functions.asp](http://www.w3schools.com/PHP/php_functions.asp)

21

# PHP Function Example

---

```
function add ($x, $y)  
{  
    $total = $x + $y;  
    return $total;  
}  
  
echo "1 + 16 = " . add (1,16);
```

1 + 16 = 17

[http://www.w3schools.com/PHP/php\\_functions.asp](http://www.w3schools.com/PHP/php_functions.asp)

22

# PHP and XML

## SimpleXML

---

- ◆ Because we want to use PHP to load XML, we need PHP 5
  - » [www.cs.odu.edu](http://www.cs.odu.edu) server has PHP 4
  - » **webspace**.cs.odu.edu server has PHP 5
- ◆ Must know the layout of the XML file
- ◆ Load XML file
  - » `$xml = simplexml_load_file ("url of xml file");`

23

## Simple XML

### Functions

---

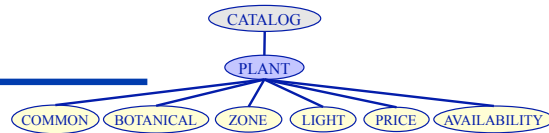
- ◆ Get the name of the first element
  - » `$xml->getName()`
- ◆ Loop through each child
  - » `foreach ($xml->children() as $child)`
    - ❖ to access label: `$child->getName()`
    - ❖ to access data: `$child`
- ◆ To get children of child
  - » `$child->children()`

24

# Simple XML

## Example

```
<?php
$xml = simplexml_load_file("plant_catalog.xml");
// access each plant
foreach ($xml->children() as $plant) {
    // access each element of the plant
    foreach ($plant->children() as $child) {
        // print label: data
        echo $child->getName() . ": " . $child . "<br />";
    }
}
?>
```



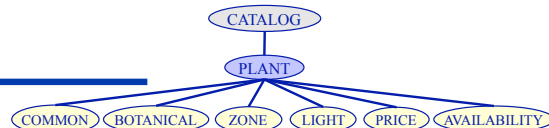
<http://webpace.cs.odu.edu/~mweigle/cs312/php/plants.php>

<http://webpace.cs.odu.edu/~mweigle/cs312/php/plants.php.txt>

25

# SimpleXML

## XPath



- ◆ After loading the xml file with SimpleXML, use XPath to find particular elements in the XML.

- ◆ Find all plants whose common names are Bloodroot

- » `$plant = $xml->xpath("//PLANT [COMMON='Bloodroot']");`
  - ❖ // - selects nodes in the document from the current node that match the selection no matter where they are
  - ❖ returns an array

- ◆ Access the price of the first plant that matches

- » `$price = (string) $plant[0]->PRICE;`

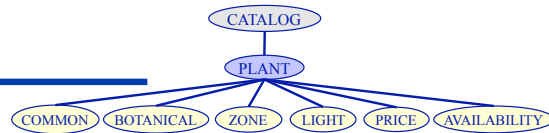
[http://www.w3schools.com/XPath/xpath\\_syntax.asp](http://www.w3schools.com/XPath/xpath_syntax.asp)

26

# XPath

## Example

---



- ◆ Find all plants with "Mostly Shady" light

```
$plant = $xml->xpath ("//PLANT [LIGHT='Mostly Shady']");
```

- ◆ Display all plants that match

```
$numPlants = count ($plant);  
for ($i=0; $i<$numPlants; $i++) {  
    echo "<p>"  
    foreach ($plant[$i]->children() as $child) {  
        echo $child->getName() . ": " . $child . "<br />";  
    }  
    echo "</p>";  
}  
} http://webspace.cs.odu.edu/~mweigle/cs312/php/plants-xpath.php  
http://webspace.cs.odu.edu/~mweigle/cs312/php/plants-xpath.php.txt
```

27

# PHP

## Outline

---



- ◆ Syntax
- ◆ Arrays
- ◆ Strings
- ◆ Loops
- ◆ Conditionals
- ◆ Functions
- ◆ Forms
- ◆ PHP and XML

28