PHP Language Intro

Reading material for PHP:
http://www.w3schools.com/php/default.asp

CS/IT 490 WD, Fall 2013

Last update 2013-09-17

Written by Rachel J. Morris

Breakdown

- What is PHP?
- Includes
- Variables and Arrays
- Echo and Print
- Useful PHP Functions
- String Manipulation
- Operators
- Control Flow

What is PHP?

- PHP is a server-side scripting language.
- Similar to other programming languages, it can open files and databases, and perform various functions.
- PHP also has a set of functions to make things easier
  - e.x.: Explode

What is PHP?

- A .php file can contain only PHP code, or it can contain a full XHTML webpage, with bits of PHP commands thrown in.
- To begin writing PHP, it must be contained within:

```
<?php /* command */ ?>
```

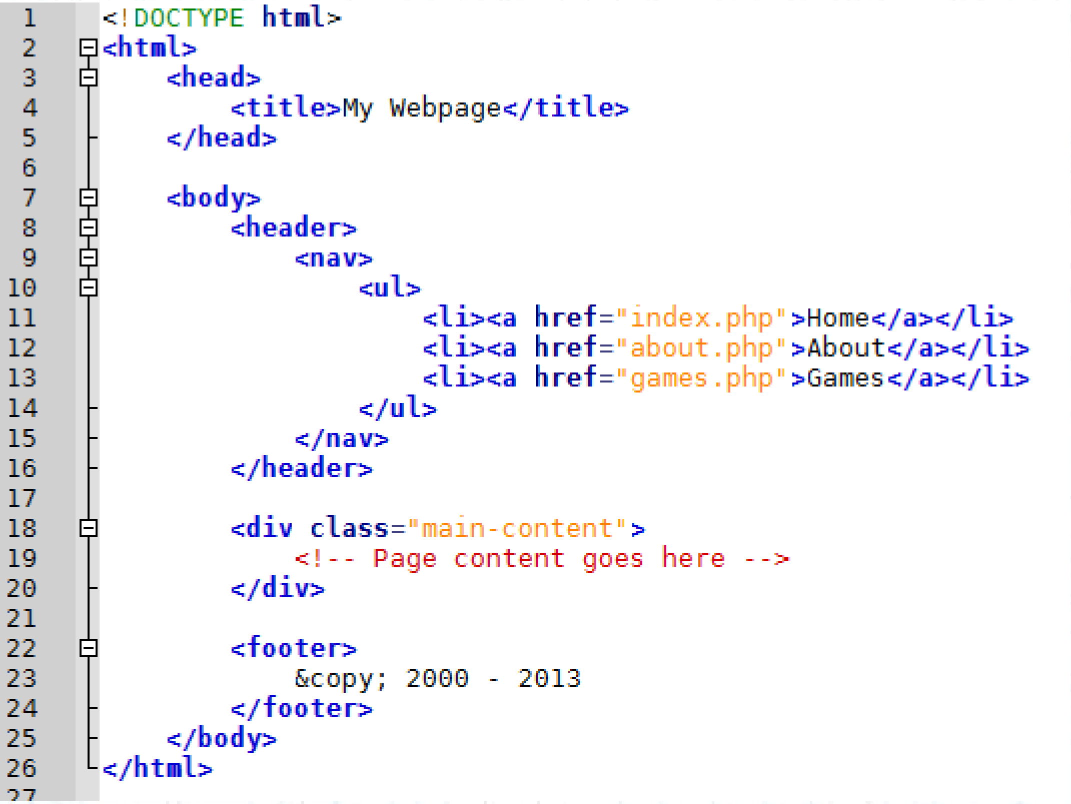
Or

```
<? /* command */ ?>
```

Includes

- Just like in Python where you can **import** other libraries and code, and C++ where you **#include** other libraries and code,
- You can use the **include** and **include_once** statements in PHP to include other .php files.
- If your .php file contains XHTML, that XHTML will be inserted into the page at the point you are using the **include**.

Includes

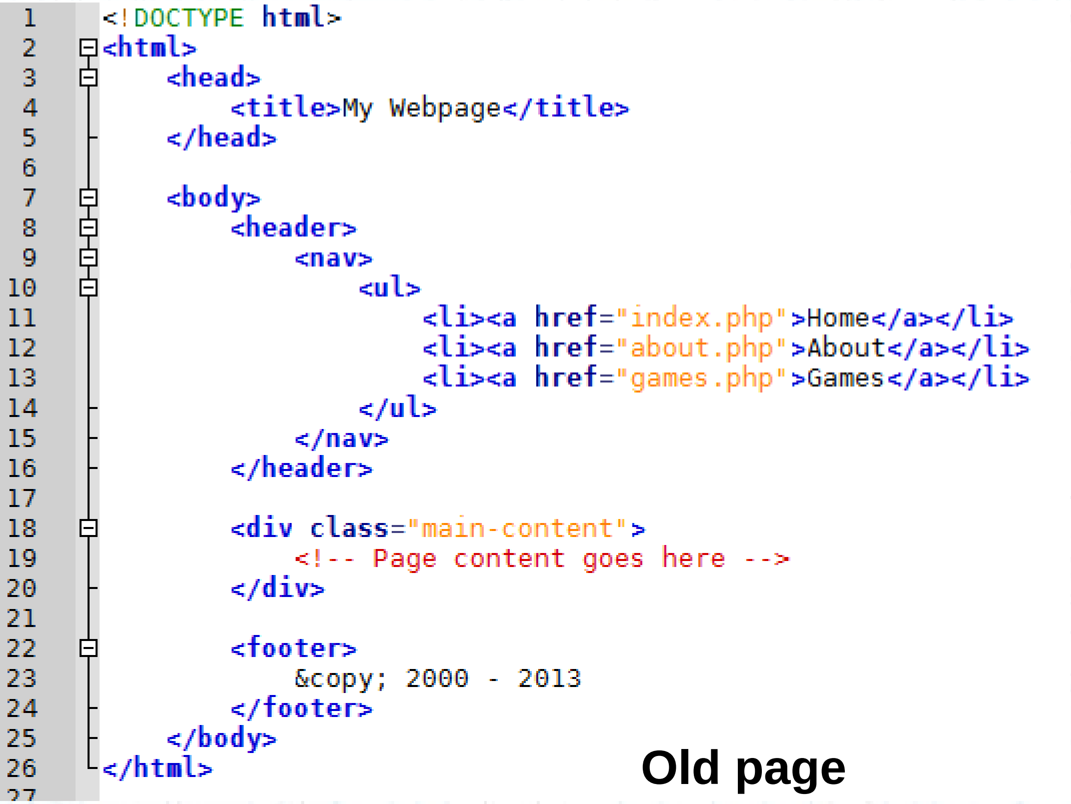


```
<!DOCTYPE html>
<html>
    <head>
        <title>My Webpage</title>
    </head>

    <body>
        <header>
            <nav>
                <ul>
                    <li><a href="index.php">Home</a></li>
                    <li><a href="about.php">About</a></li>
                    <li><a href="games.php">Games</a></li>
                </ul>
            </nav>
        </header>

        <div class="main-content">
            <!-- Page content goes here -->
        </div>

        <footer>
            &copy; 2000 - 2013
        </footer>
    </body>
</html>
```

A full webpage, with the required XHTML tags, a <header>, <div class=”main-content”>, and <footer>.

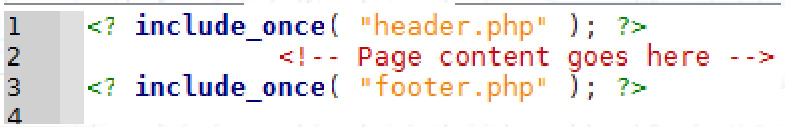
We can split up the three main sections...

Includes



```
<!DOCTYPE html>
<html>
    <head>
        <title>My Webpage</title>
    </head>

    <body>
        <header>
            <nav>
                <ul>
                    <li><a href="index.php">Home</a></li>
                    <li><a href="about.php">About</a></li>
                    <li><a href="games.php">Games</a></li>
                </ul>
            </nav>
        </header>

        <div class="main-content">
            <!-- Page content goes here -->
        </div>

        <footer>
            &copy; 2000 - 2013
        </footer>
    </body>
</html>
```

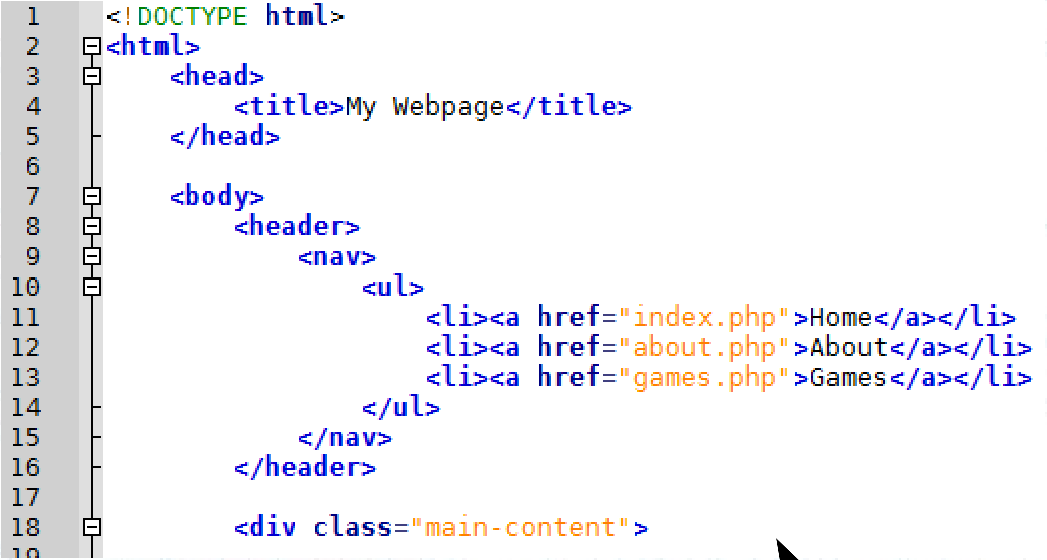
Old page



```
<? include_once( "header.php" ); ?>
            <!-- Page content goes here -->
<? include_once( "footer.php" ); ?>
```

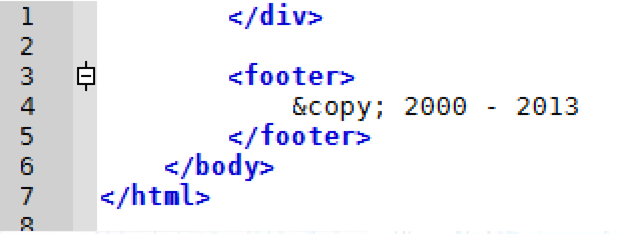
The new page only contains this

Includes



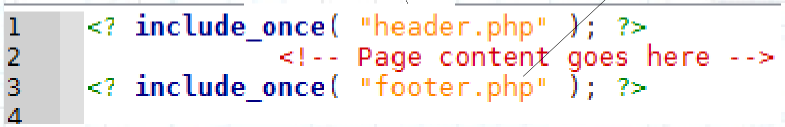
```
<!DOCTYPE html>
<html>
    <head>
        <title>My Webpage</title>
    </head>

    <body>
        <header>
            <nav>
                <ul>
                    <li><a href="index.php">Home</a></li>
                    <li><a href="about.php">About</a></li>
                    <li><a href="games.php">Games</a></li>
                </ul>
            </nav>
        </header>

        <div class="main-content">
```

header.php



```
        </div>

        <footer>
            &copy; 2000 - 2013
        </footer>
    </body>
</html>
```

footer.php



```
<? include_once( "header.php" ); ?>
            <!-- Page content goes here -->
<? include_once( "footer.php" ); ?>
```

index.php

Includes

- Generally, use include_once instead of include.
- This ensures that if a file is included multiple times, the only time it's copied over is the first time.
- This keeps “duplicate” variables, functions, and classes from being declared.

Variables and Arrays

- Variables in PHP start with a dollar sign: $

```
<? $username = "bob"; ?>
```

- Arrays must be first declared like this:

```
<? $userInfo = Array(); ?>
```

Written by Rachel J. Morris

Variables and Arrays

- You don't need to declare a variable before using it, and you don't need to specify a data-type.
- But there are still different *datatypes*:
  - Strings: enclosed in double- or single-quotes.

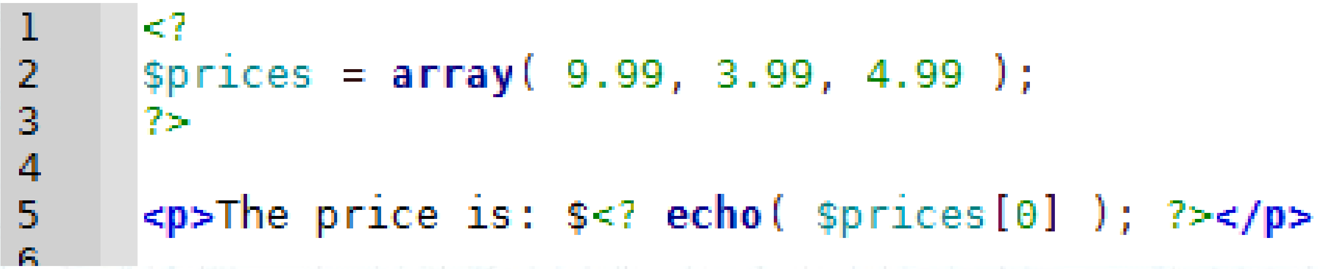
```
<? $url = "http://www.asdf.com"; ?>
<? $name = 'claire'; ?>
```

Variables and Arrays

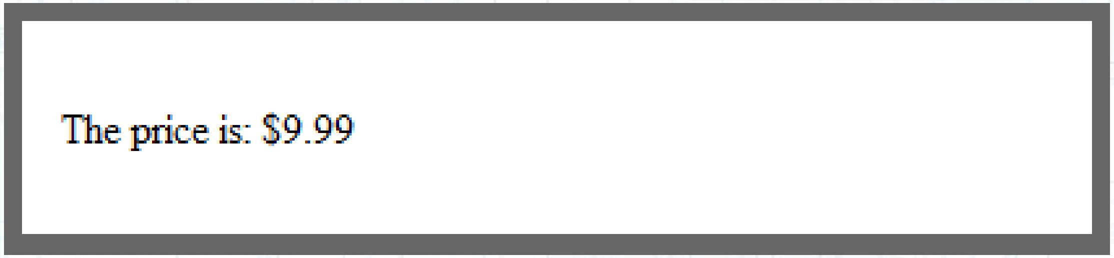
- Integers:
```
<? $age = 54; ?>
```
- Floats:
```
<? $price = 19.95; ?>
```
- Booleans:
```
<? $isLoggedIn = false; ?>
```
- NULL:
```
<? $userLogin = null; ?>
```
- Objects:
  - We can declare and instantiate classes in PHP, which we will go over later.

Variables and Arrays

- Arrays are Associative Arrays
- If you just store a series of values, the keys are the indices in order:



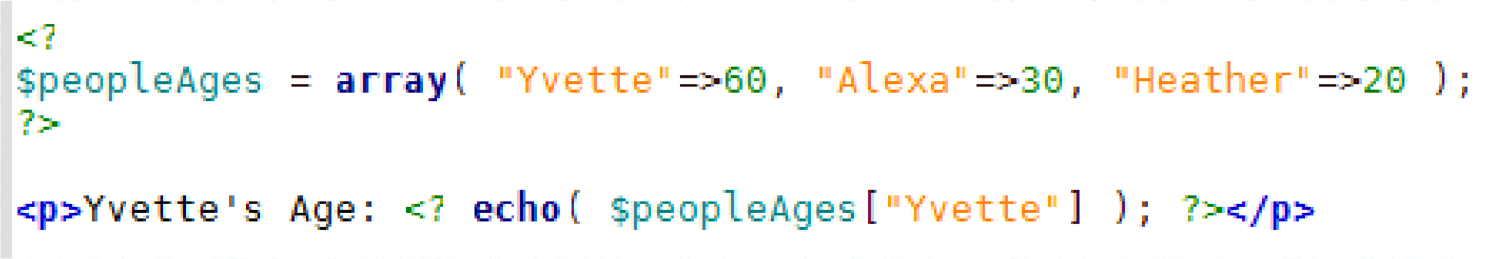
```
<?
$prices = array( 9.99, 3.99, 4.99 );
?>

<p>The price is: $<? echo( $prices[0] ); ?></p>
```



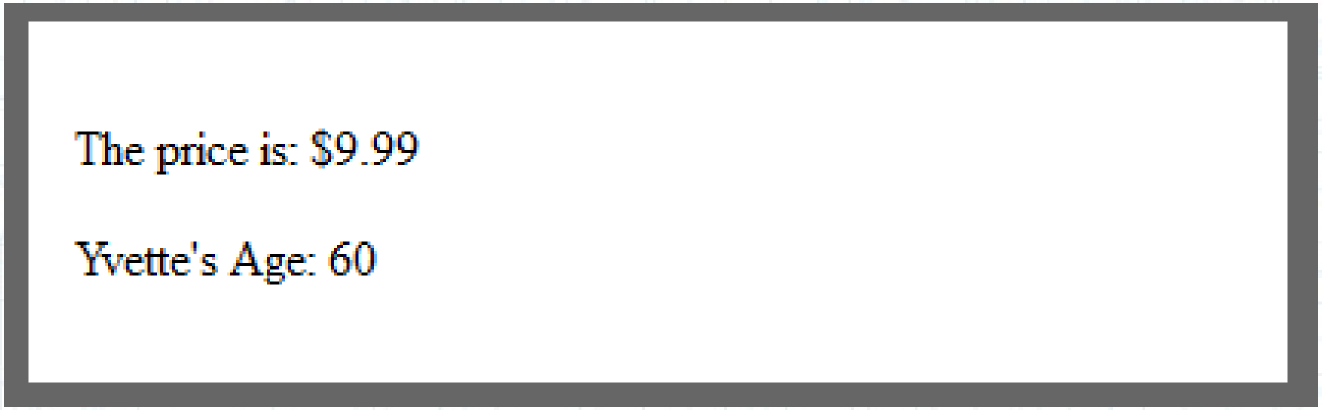
```
The price is: $9.99
```

Variables and Arrays

- You can also specify the key-value pairs:



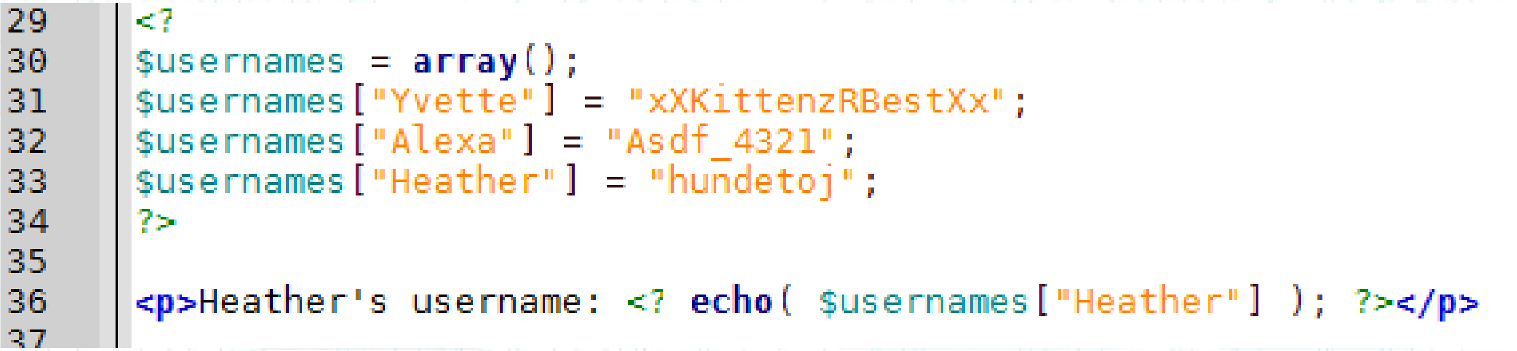
```
<?
$peopleAges = array( "Yvette"=>60, "Alexa"=>30, "Heather"=>20 );
?>

<p>Yvette's Age: <? echo( $peopleAges["Yvette"] ); ?></p>
```



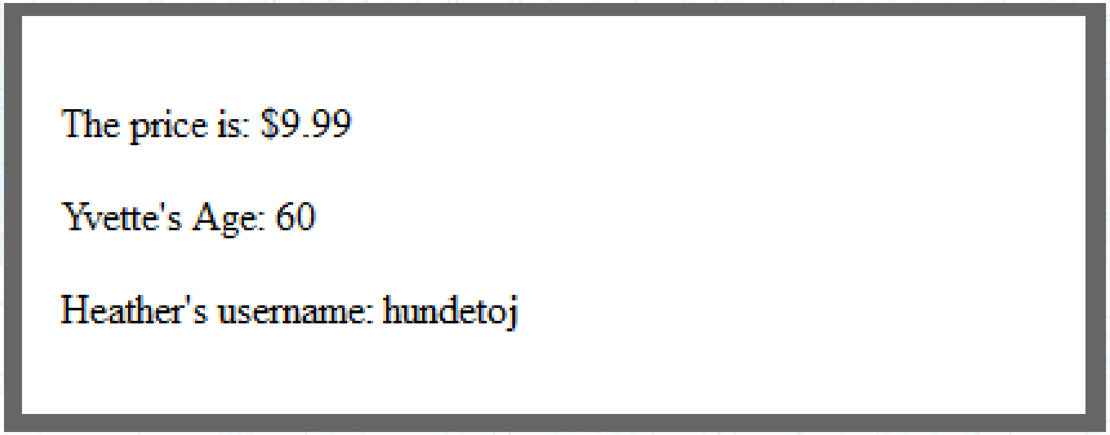
```
The price is: $9.99

Yvette's Age: 60
```

Variables and Arrays

- Another way to declare a key-value pair:



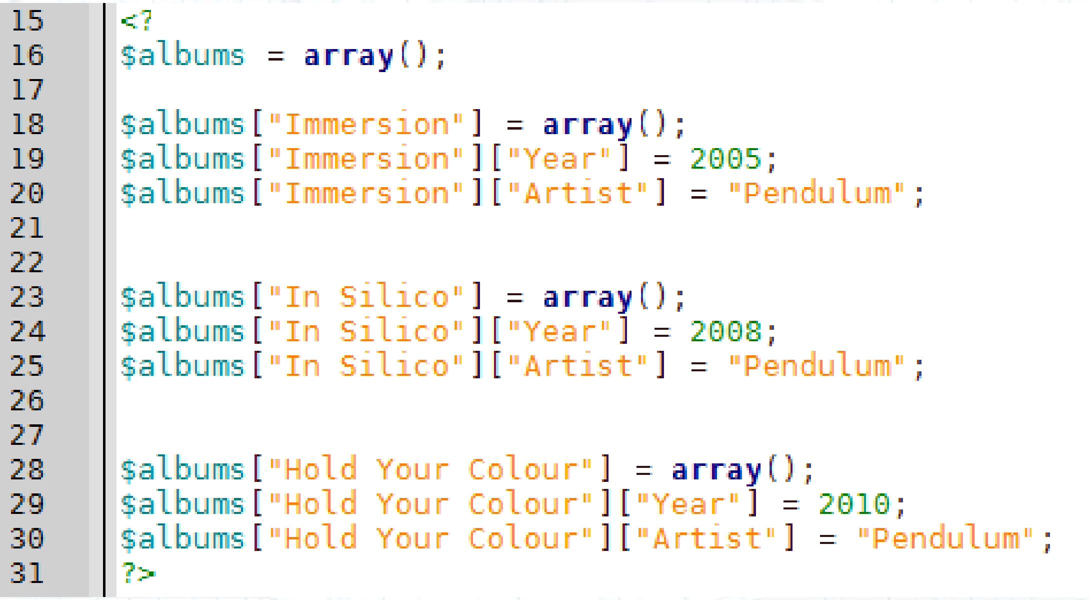
```
<?
$usernames = array();
$usernames["Yvette"] = "xXKittenzRBestXx";
$usernames["Alexa"] = "Asdf_4321";
$usernames["Heather"] = "hundetoj";
?>

<p>Heather's username: <? echo( $usernames["Heather"] ); ?></p>
```



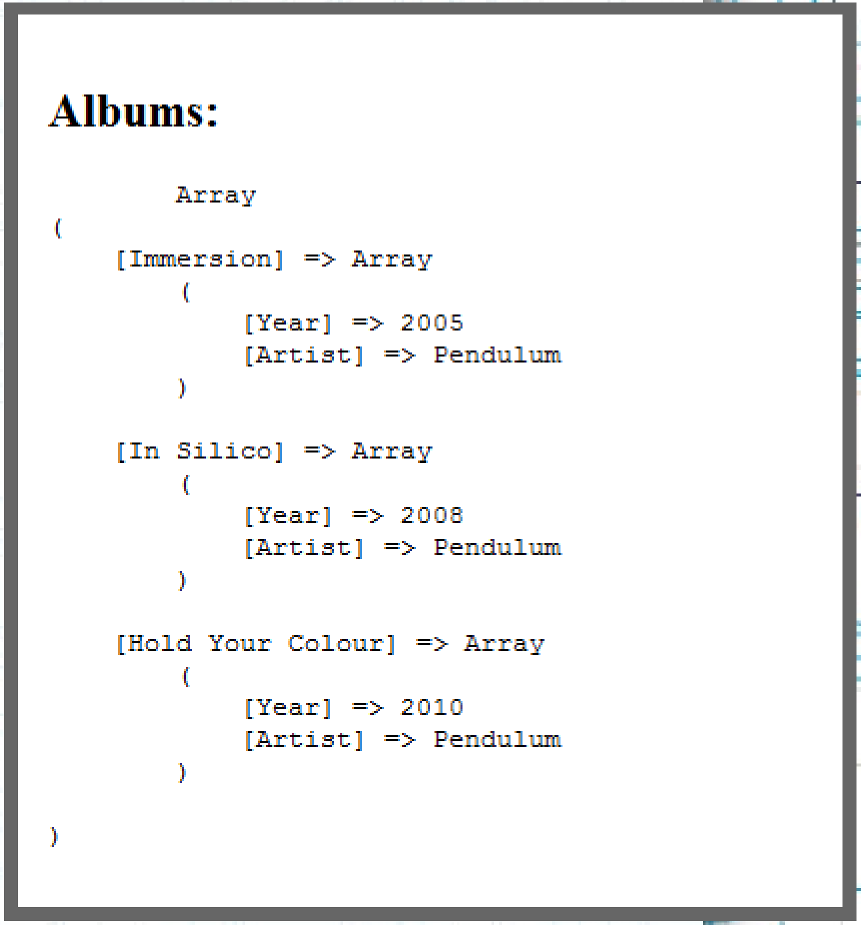
```
The price is: $9.99

Yvette's Age: 60

Heather's username: hundetoj
```

Variables and Arrays

- Array values can also contain additional arrays, to make a hierarchy:



```php
<?
$albums = array();

$albums["Immersion"] = array();
$albums["Immersion"]["Year"] = 2005;
$albums["Immersion"]["Artist"] = "Pendulum";


$albums["In Silico"] = array();
$albums["In Silico"]["Year"] = 2008;
$albums["In Silico"]["Artist"] = "Pendulum";


$albums["Hold Your Colour"] = array();
$albums["Hold Your Colour"]["Year"] = 2010;
$albums["Hold Your Colour"]["Artist"] = "Pendulum";
?>
```

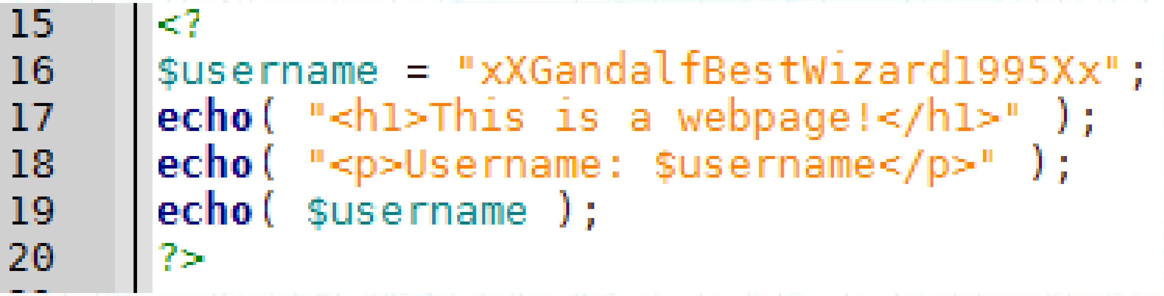


Albums:

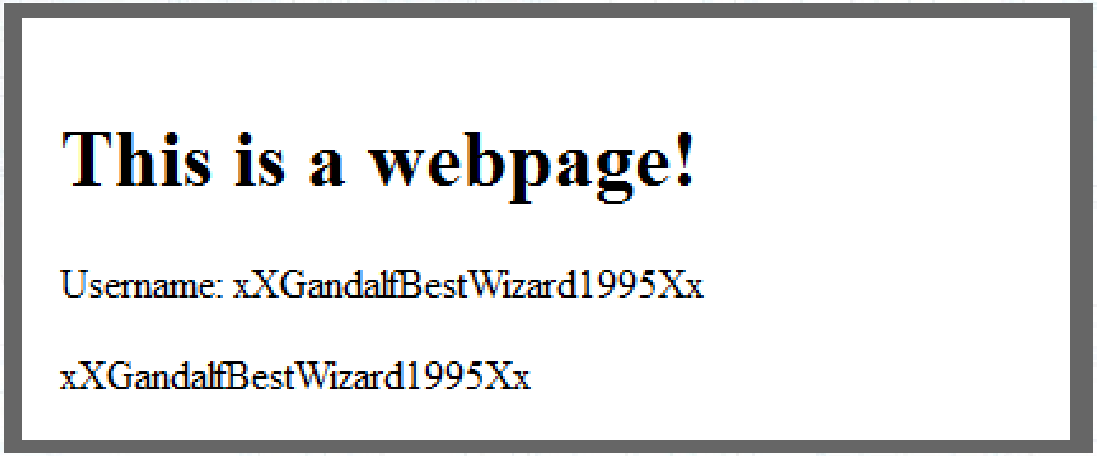
```
        Array
(
    [Immersion] => Array
        (
            [Year] => 2005
            [Artist] => Pendulum
        )

    [In Silico] => Array
        (
            [Year] => 2008
            [Artist] => Pendulum
        )

    [Hold Your Colour] => Array
        (
            [Year] => 2010
            [Artist] => Pendulum
        )

)
```

Echo and Print

- If you want to print text into the HTML page through PHP, you can use **echo**:



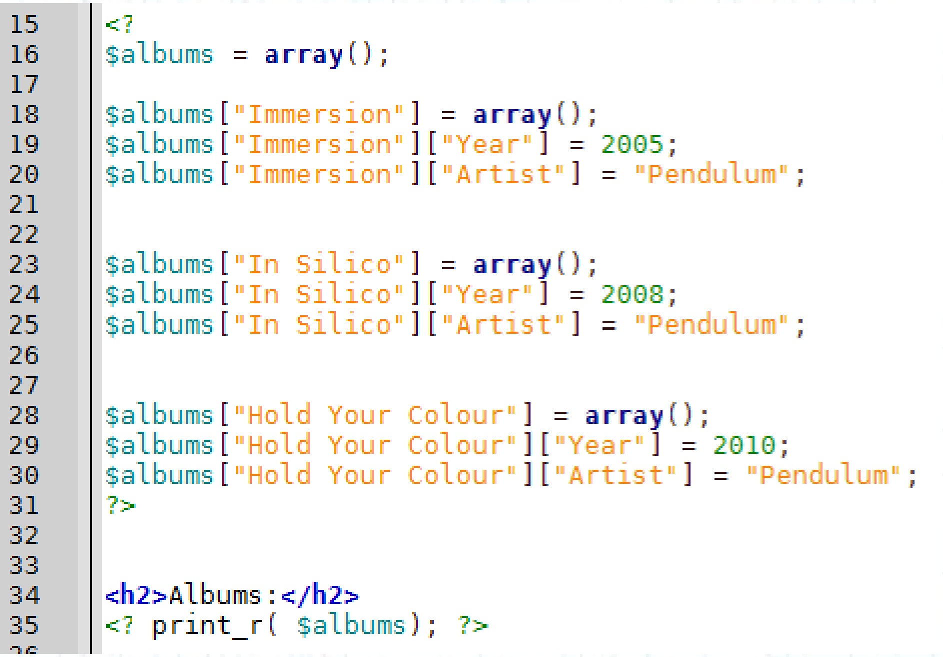
```
<?
$username = "xXGandalfBestWizard1995Xx";
echo( "<h1>This is a webpage!</h1>" );
echo( "<p>Username: $username</p>" );
echo( $username );
?>
```



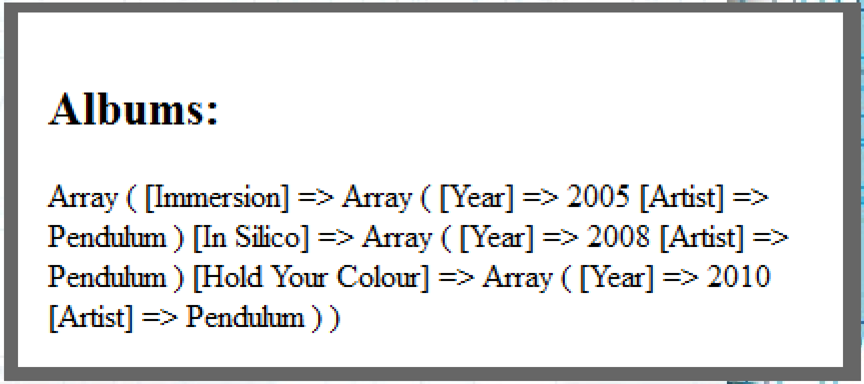
```
This is a webpage!

Username: xXGandalfBestWizard1995Xx

xXGandalfBestWizard1995Xx
```

Echo and Print

- If you want to print out all the contents of an array, you can use **print_r**:



```
<?
$albums = array();

$albums["Immersion"] = array();
$albums["Immersion"]["Year"] = 2005;
$albums["Immersion"]["Artist"] = "Pendulum";


$albums["In Silico"] = array();
$albums["In Silico"]["Year"] = 2008;
$albums["In Silico"]["Artist"] = "Pendulum";


$albums["Hold Your Colour"] = array();
$albums["Hold Your Colour"]["Year"] = 2010;
$albums["Hold Your Colour"]["Artist"] = "Pendulum";
?>


<h2>Albums:</h2>
<? print_r( $albums); ?>
```

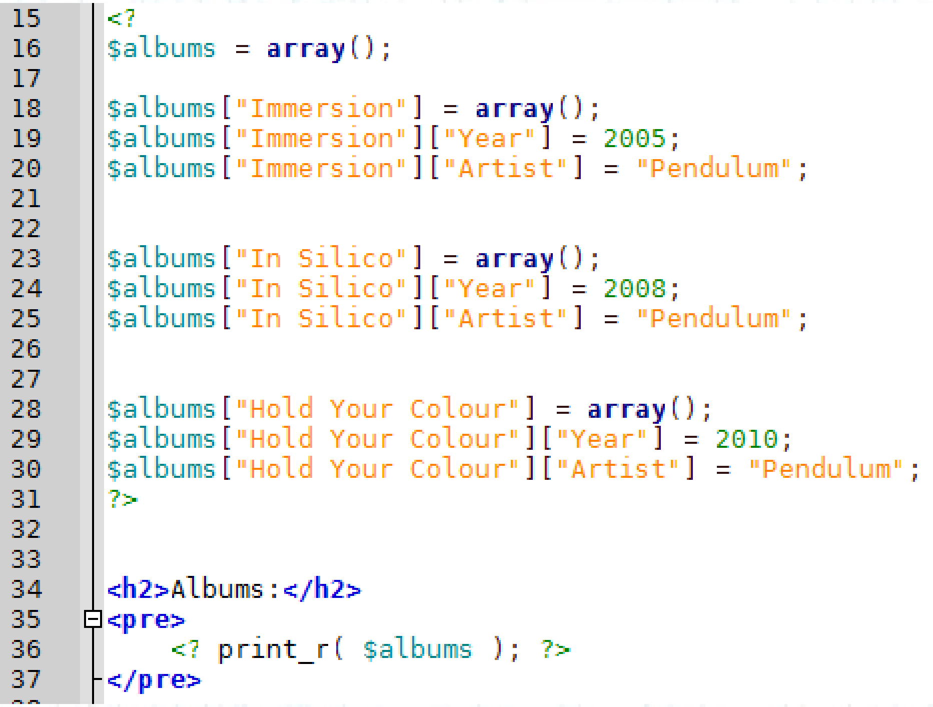


Albums:

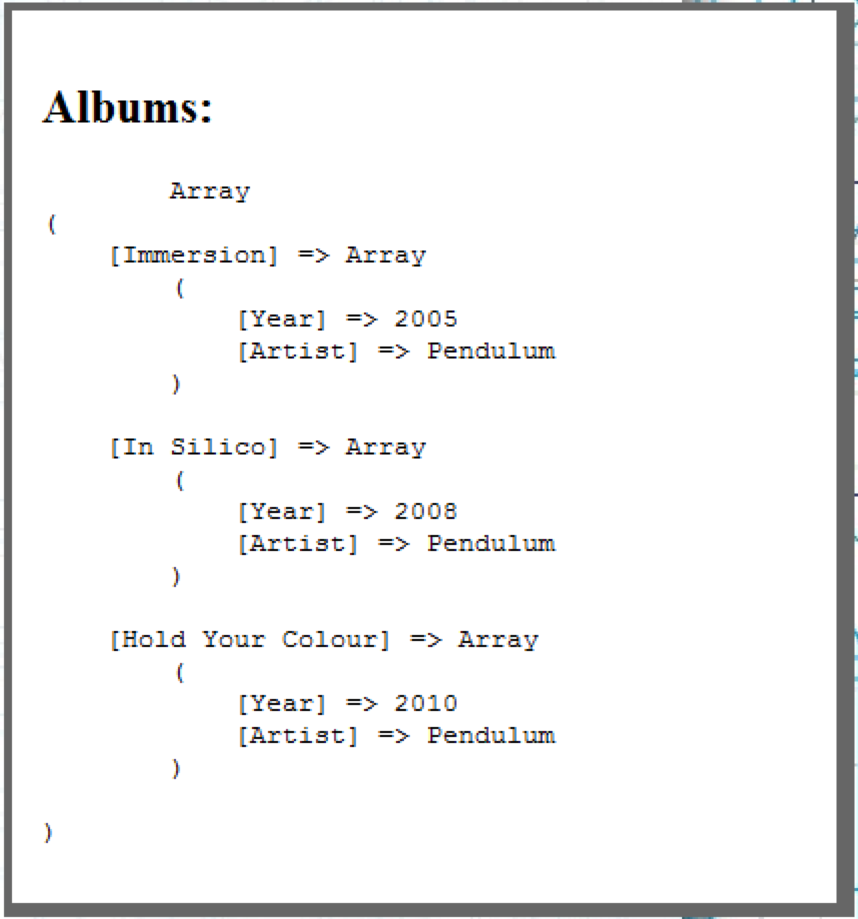
```
Array ( [Immersion] => Array ( [Year] => 2005 [Artist] =>
Pendulum ) [In Silico] => Array ( [Year] => 2008 [Artist] =>
Pendulum ) [Hold Your Colour] => Array ( [Year] => 2010
[Artist] => Pendulum ) )
```

Echo and Print

- Tip: The array printout looks nicer if you enclose it within <pre> tags. This can be useful for debuggng.



```
<?
$albums = array();

$albums["Immersion"] = array();
$albums["Immersion"]["Year"] = 2005;
$albums["Immersion"]["Artist"] = "Pendulum";


$albums["In Silico"] = array();
$albums["In Silico"]["Year"] = 2008;
$albums["In Silico"]["Artist"] = "Pendulum";


$albums["Hold Your Colour"] = array();
$albums["Hold Your Colour"]["Year"] = 2010;
$albums["Hold Your Colour"]["Artist"] = "Pendulum";
?>


<h2>Albums:</h2>
<pre>
    <? print_r( $albums ); ?>
</pre>
```

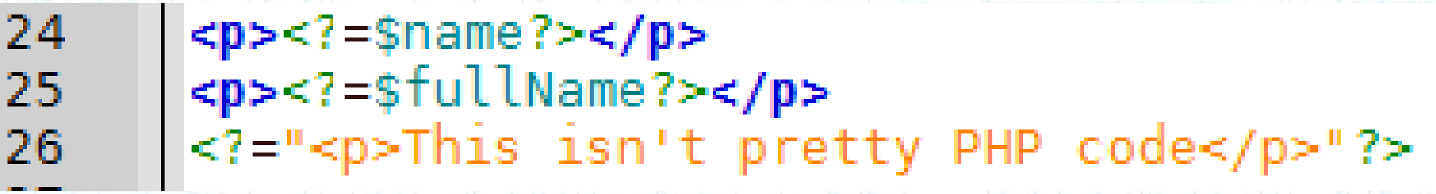


Albums:

```
        Array
(
    [Immersion] => Array
        (
            [Year] => 2005
            [Artist] => Pendulum
        )

    [In Silico] => Array
        (
            [Year] => 2008
            [Artist] => Pendulum
        )

    [Hold Your Colour] => Array
        (
            [Year] => 2010
            [Artist] => Pendulum
        )

)
```

Echo and Print

- The short-hand for Echo is to use an equal sign =

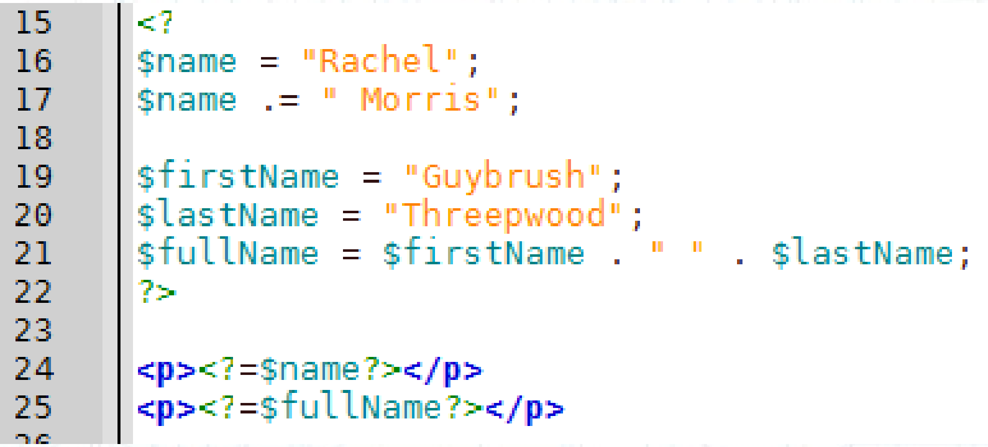


```
<p><?=$name?></p>
<p><?=$fullName?></p>
<?="<p>This isn't pretty PHP code</p>"?>
```

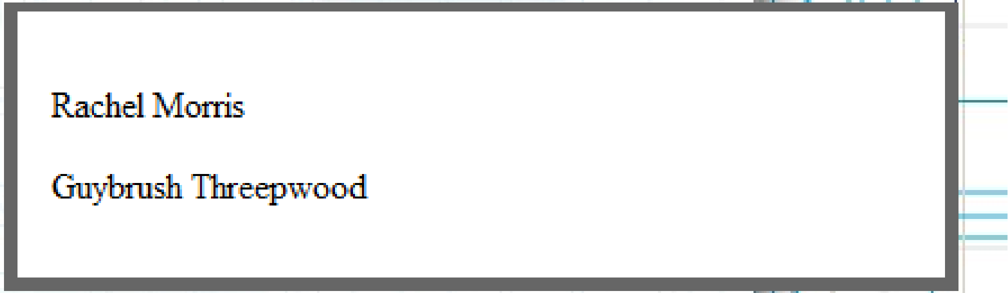
- Good for subbing in names within your HTML pages.

String Manipulation

- In PHP, to concatenate two strings you use the period . Operator.
    - Don't get it mixed up with +! + will not concatenate your strings!



```
<?
$name = "Rachel";
$name .= " Morris";

$firstName = "Guybrush";
$lastName = "Threepwood";
$fullName = $firstName . " " . $lastName;
?>

<p><?=$name?></p>
<p><?=$fullName?></p>
```



```
Rachel Morris

Guybrush Threepwood
```

String Manipulation

- PHP has some handy functions for you to play with strings with:

explode($delimiter, $string)
Splits a string by the delimiter specified
http://php.net/manual/en/function.explode.php

str_getcsv()
Turns a .csv string into an array
http://www.w3schools.com/php/func_string_str_getcsv.asp

str_replace($search, $replace, $text)
Replaces any occurrences of [$search] with [$replace]
http://www.php.net/manual/en/function.str-replace.php

String Manipulation

strtoupper($string) and strtolower($string)
Set to upper/lower case
http://www.php.net/manual/en/function.strtoupper.php
http://www.php.net/manual/en/function.strtolower.php

trim($string)
Removes any whitespace before the first character and after the last character.
http://www.php.net/manual/en/function.trim.php

strpos($haystack, $needle)
Find the position of a substring
http://php.net/manual/en/function.strpos.php

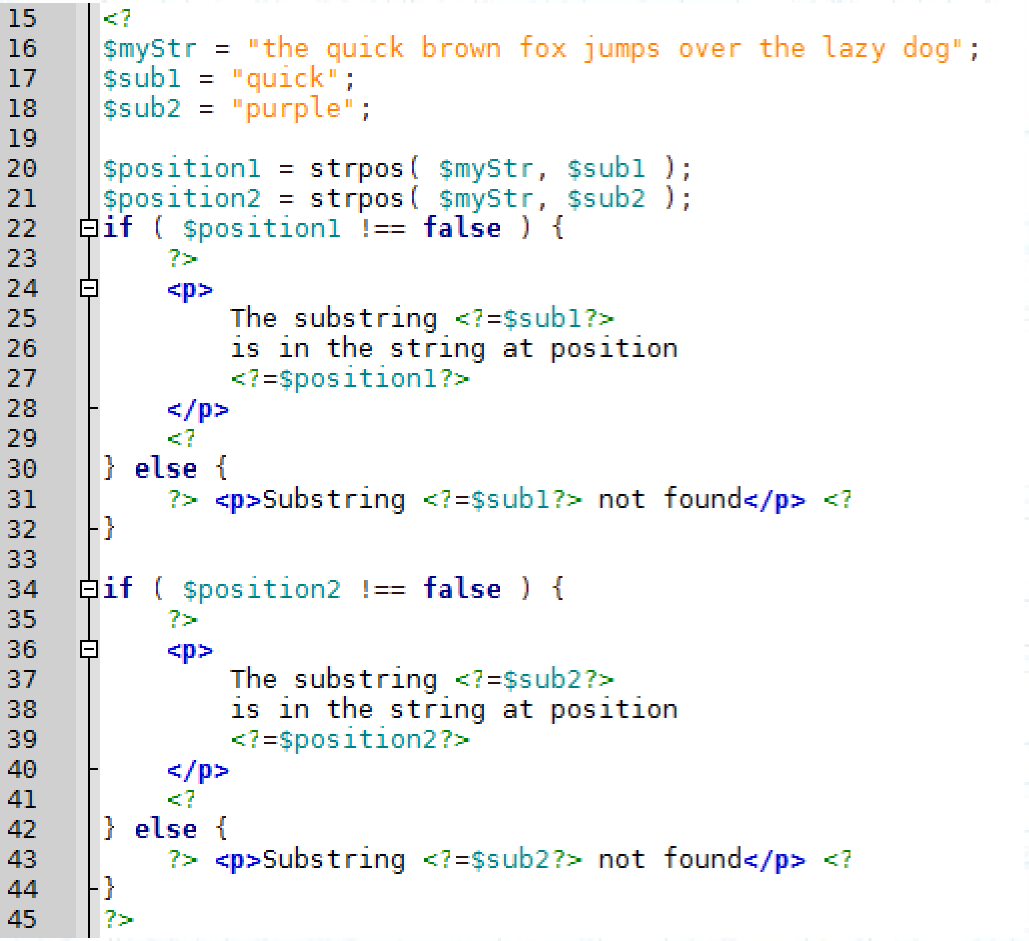
String Manipulation

- The problem with strpos
  - It returns the **position index** of a substring.
  - What does it return if that substring is not found?
  - False = 0, but that's also a position!

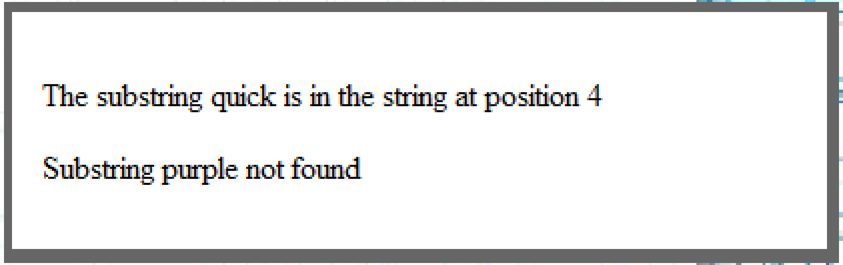
String Manipulation

- The problem with strpos
  - To check if a sub-string (needle) is not in the string (haystack), you need to use the === or !== operators:

String Manipulation



```
<?
$myStr = "the quick brown fox jumps over the lazy dog";
$sub1 = "quick";
$sub2 = "purple";

$position1 = strpos( $myStr, $sub1 );
$position2 = strpos( $myStr, $sub2 );
if ( $position1 !== false ) {
    ?>
    <p>
        The substring <?=$sub1?>
        is in the string at position
        <?=$position1?>
    </p>
    <?
} else {
    ?> <p>Substring <?=$sub1?> not found</p> <?
}

if ( $position2 !== false ) {
    ?>
    <p>
        The substring <?=$sub2?>
        is in the string at position
        <?=$position2?>
    </p>
    <?
} else {
    ?> <p>Substring <?=$sub2?> not found</p> <?
}
?>
```



```
The substring quick is in the string at position 4

Substring purple not found
```

String Manipulation

- We will also talk about the tag & slash stripping functions when we get to **string sanitation**.

Operators

- PHP has your standard math operators:

Operator	Example
Addition +	$num1 + $num2
Subtraction -	$num1 - $num2
Multiplication *	$num1 * $num2
Division /	$num1 / $num2
Modulus %	$num1 % $num2

- You can also use +=, -=, *=, /=, etc.

Operators

- The assignment operator is the same as C++:
  $x = 3
- Concatenation uses a period .
  $employee .= " (fired)";
- Increment and decrement are like C++:
  $i++; ++$i; $i--; --$i;

http://www.w3schools.com/php/php_operators.asp

Operators

- Equality operator: ==
- Identical operator: ===
  This will return **true** if the left hand equals the right hand, and they are the same type.
- Not Equal: != and <>
- Not Identical: !==

http://www.w3schools.com/php/php_operators.asp

Operators

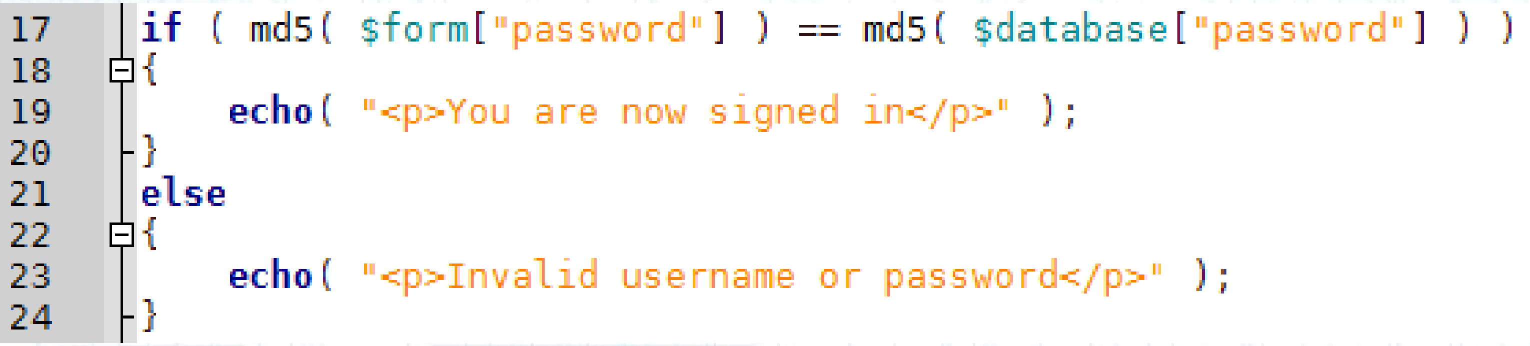
- Comparison: < <= > >=
- Logical operators:
  && or **and**
  || or **or**
  ! not

http://www.w3schools.com/php/php_operators.asp

Written by Rachel J. Morris

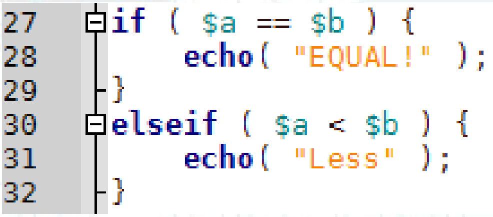
Control Flow

- PHP has your standard if / else / else if statements:



```
if ( md5( $form["password"] ) == md5( $database["password"] ) )
{
    echo( "<p>You are now signed in</p>" );
}
else
{
    echo( "<p>Invalid username or password</p>" );
}
```

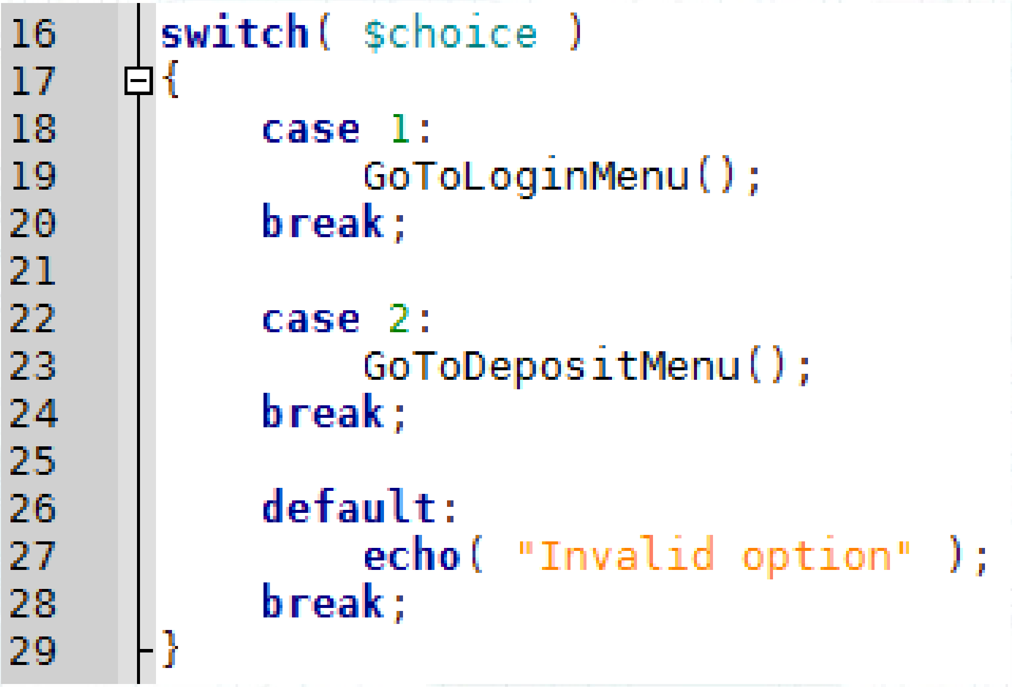
Else if is written as **elseif** though:



```
if ( $a == $b ) {
    echo( "EQUAL!" );
}
elseif ( $a < $b ) {
    echo( "Less" );
}
```

Control Flow

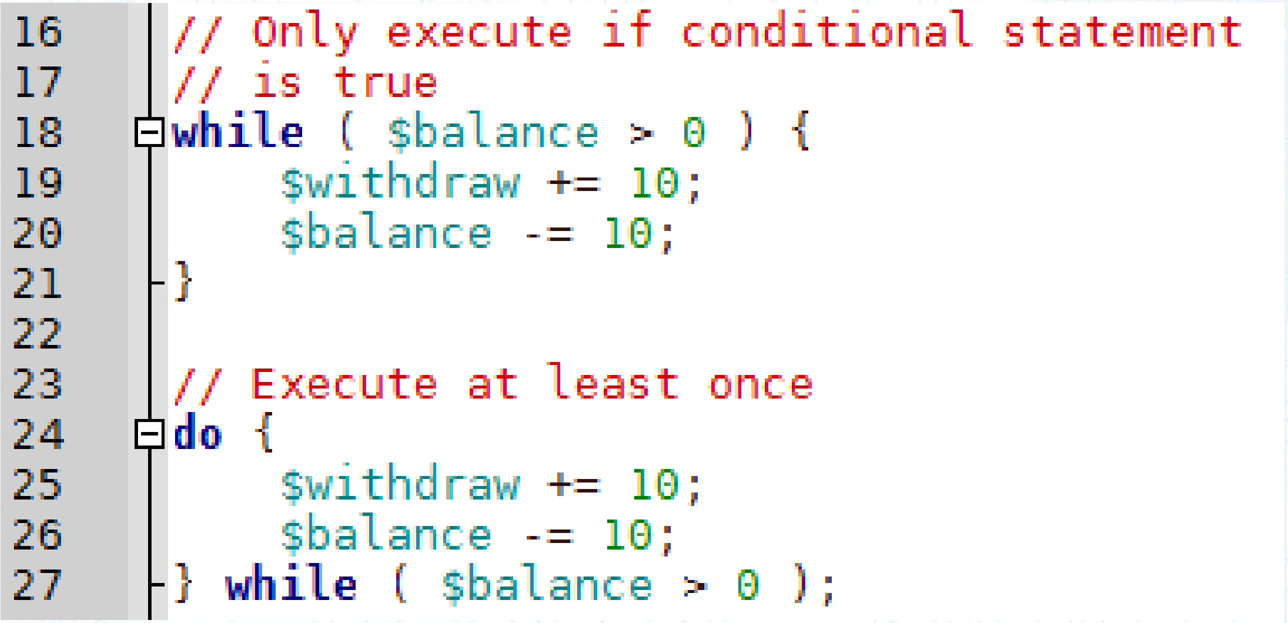
- The Switch statement is available:



```
switch( $choice )
{
    case 1:
        GoToLoginMenu();
    break;

    case 2:
        GoToDepositMenu();
    break;

    default:
        echo( "Invalid option" );
    break;
}
```

Control Flow

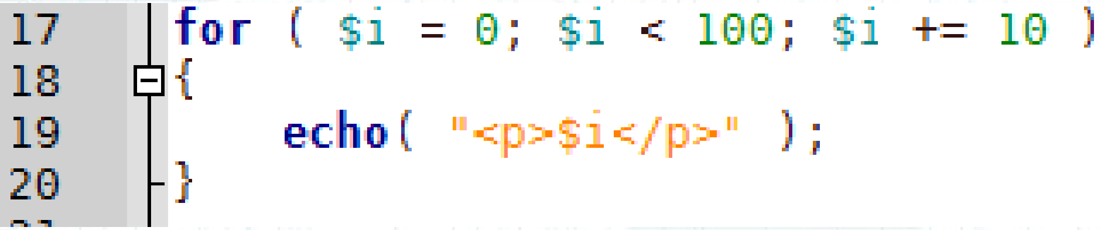
- While and Do While loops:



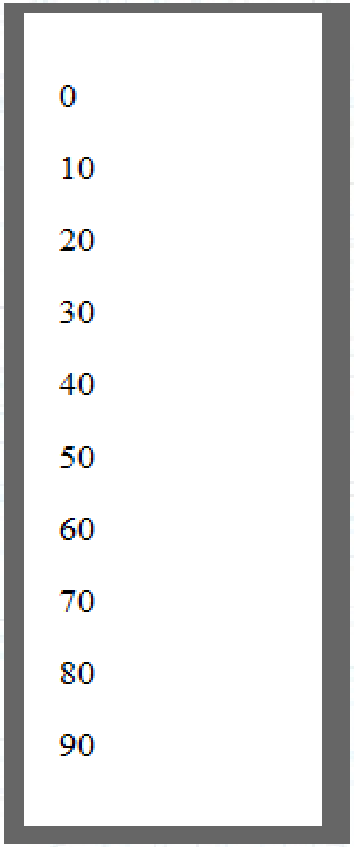
```
// Only execute if conditional statement
// is true
while ( $balance > 0 ) {
	$withdraw += 10;
	$balance -= 10;
}

// Execute at least once
do {
	$withdraw += 10;
	$balance -= 10;
} while ( $balance > 0 );
```

Control Flow

- For Loops:



```
for ( $i = 0; $i < 100; $i += 10 )
{
    echo( "<p>$i</p>" );
}
```

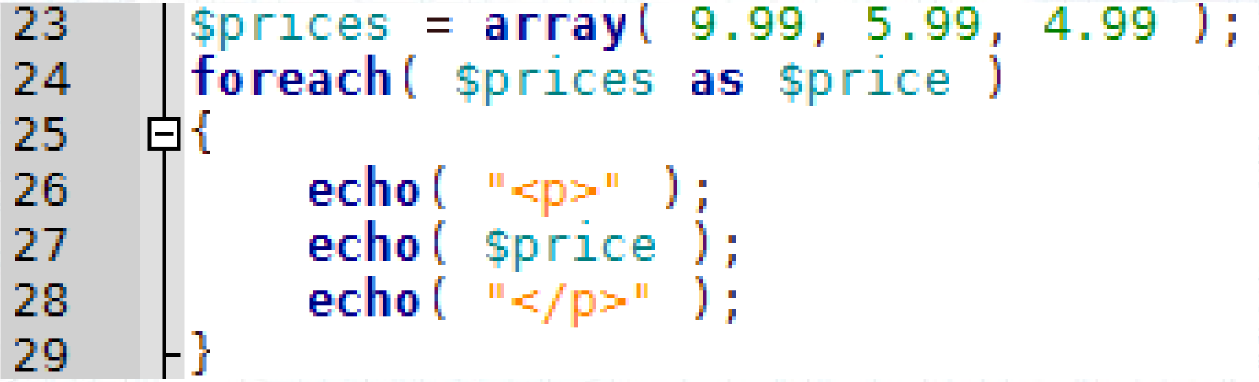


```
0

10

20

30

40

50

60

70

80

90
```

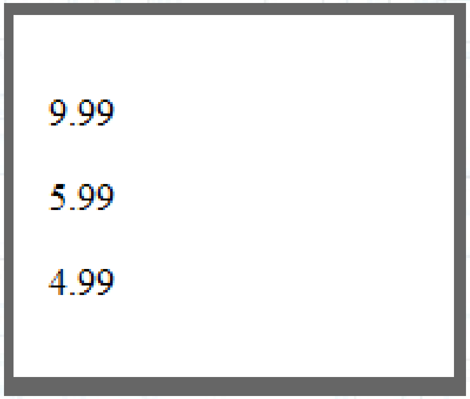
Control Flow

- For Each loops

See also http://www.w3schools.com/php/php_looping_for.asp



```php
$prices = array( 9.99, 5.99, 4.99 );
foreach( $prices as $price )
{
    echo( "<p>" );
    echo( $price );
    echo( "</p>" );
}
```



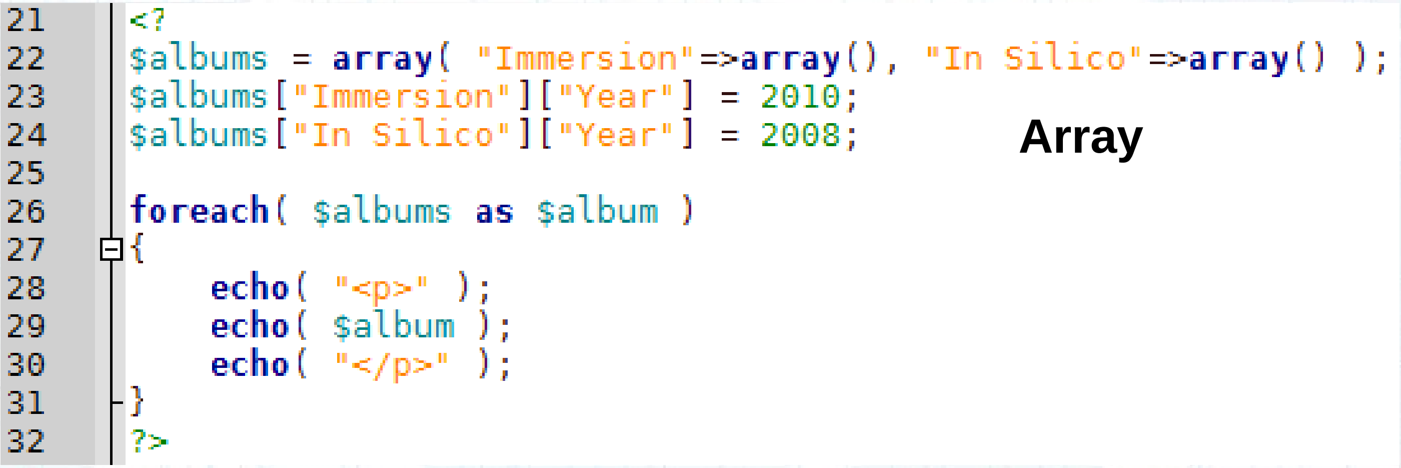
```
9.99

5.99

4.99
```

Lets us signify each element of the array with the $price variable name, so we don't have to mess with indices.

Control Flow

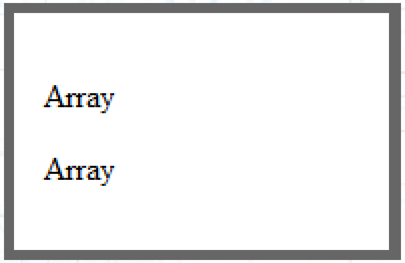
- For Each loops

See also http://www.w3schools.com/php/php_looping_for.asp



```php
<?
$albums = array( "Immersion"=>array(), "In Silico"=>array() );
$albums["Immersion"]["Year"] = 2010;
$albums["In Silico"]["Year"] = 2008;

foreach( $albums as $album )
{
    echo( "<p>" );
    echo( $album );
    echo( "</p>" );
}
?>
```

Array

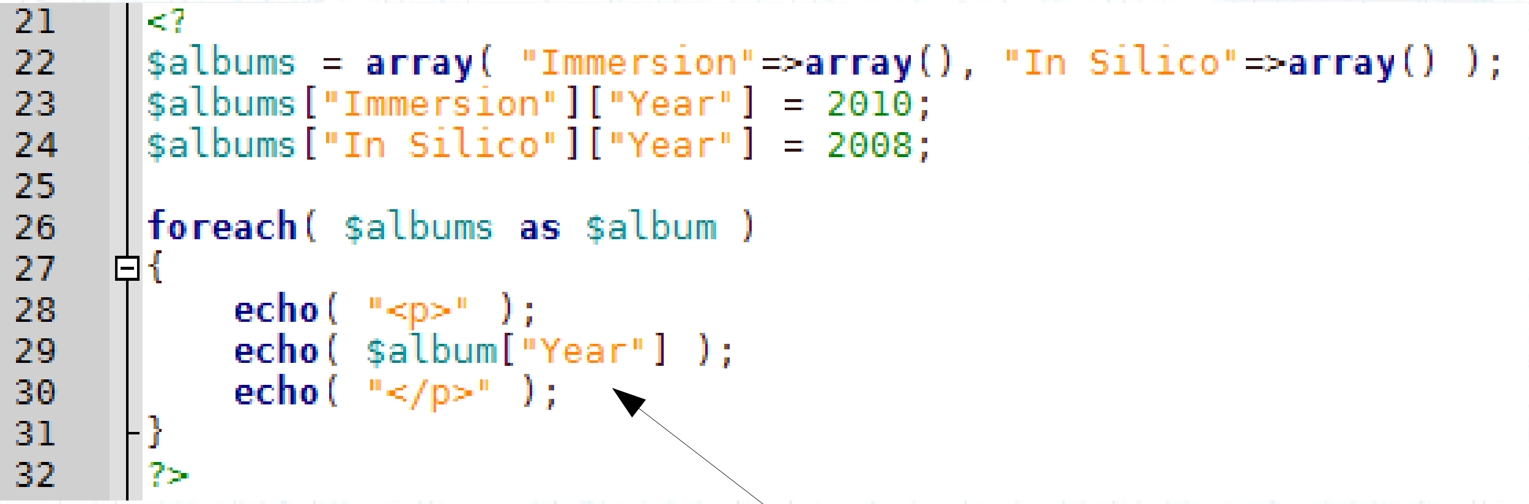


```
Array

Array
```

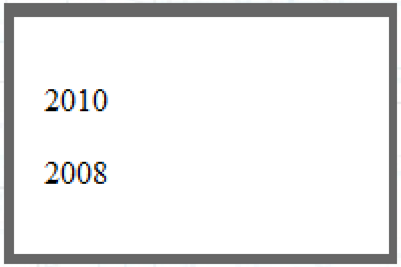
Notice that echo just shows “Array”. You can print arrays with print_r.

Control Flow

- For Each loops
  See also http://www.w3schools.com/php/php_looping_for.asp



```php
<?
$albums = array( "Immersion"=>array(), "In Silico"=>array() );
$albums["Immersion"]["Year"] = 2010;
$albums["In Silico"]["Year"] = 2008;

foreach( $albums as $album )
{
    echo( "<p>" );
    echo( $album["Year"] );
    echo( "</p>" );
}
?>
```



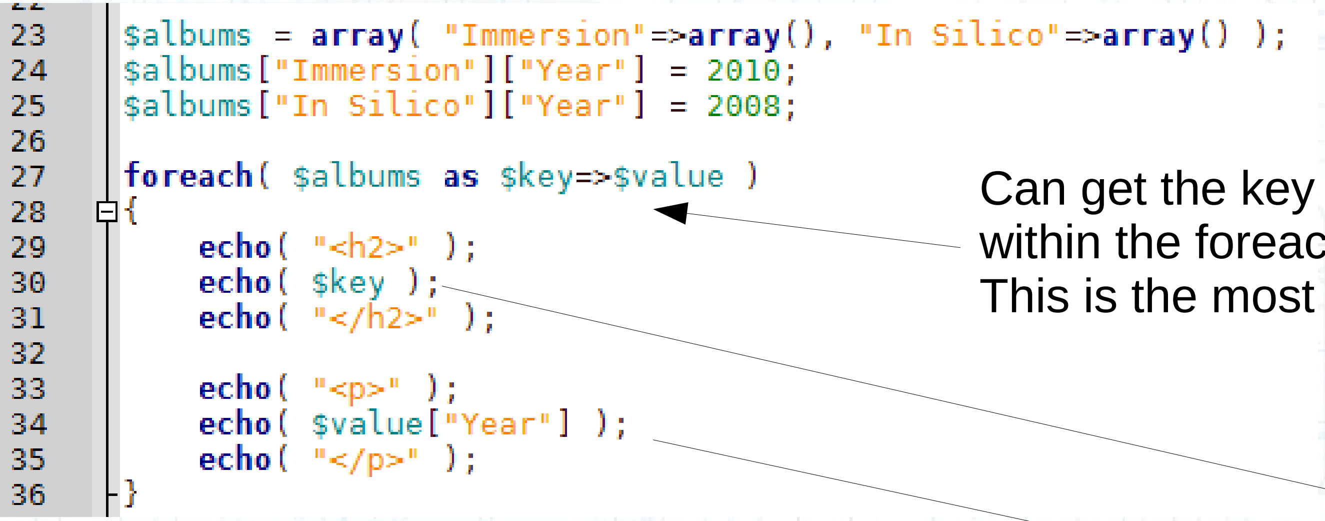
```
2010

2008
```

We can access members of the sub arrays

Control Flow

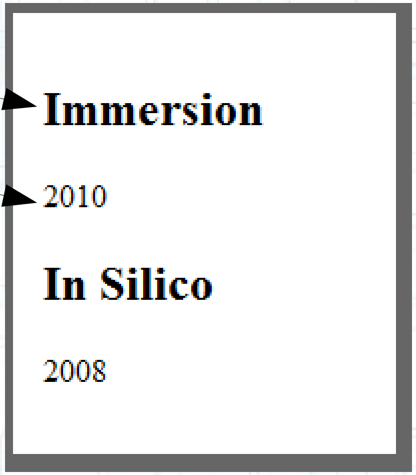
- For Each loops

  See also http://www.w3schools.com/php/php_looping_for.asp



```php
$albums = array( "Immersion"=>array(), "In Silico"=>array() );
$albums["Immersion"]["Year"] = 2010;
$albums["In Silico"]["Year"] = 2008;

foreach( $albums as $key=>$value )
{
    echo( "<h2>" );
    echo( $key );
    echo( "</h2>" );

    echo( "<p>" );
    echo( $value["Year"] );
    echo( "</p>" );
}
```

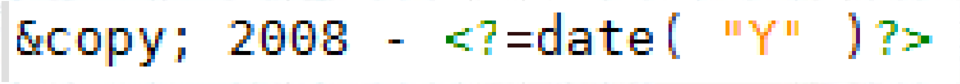
Can get the key and value within the foreach loop. This is the most handy



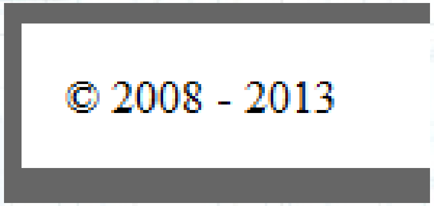
```
Immersion

2010

In Silico

2008
```

Useful PHP Functions

- You can print out the date with the date() function.
- You supply a string, which has symbols to represent what format we want for our timestamp.
- See http://www.w3schools.com/php/php_date.asp for the different options.



```
&copy; 2008 - <?=date( "Y" )?>
```



© 2008 - 2013

Useful PHP Functions

- More later...

References

-