



Universiteti i Prishtinës

ADVANCED PHP CONCEPTS

Slides mostly adopted from Tufts University, Prof. Ming Chow

Dr. Ing. Lule Ahmedi
Lënda: Programimi në Internet

SimpleXML

- Simple way to parse XML files (unless your XML document has namespaces (e.g., tags with "colons"))
- Requires PHP 5
- Complete list of SimpleXML functions at:
<http://us3.php.net/manual/en/ref.simplexml.php>

SimpleXML: Example 1

- Loading an XML file and print out elements

```
<?php
$xml = simplexml_load_file('twitter.xml');
print_r($xml);
echo "***** End of printing entire XML object *****\n";
foreach ($xml as $item) {
    print_r($item);
    echo "***** End of printing item id = " . $item->id . "
*****\n";
    if ($item instanceof SimpleXMLElement) {
        echo "***** item is an instance of SimpleXMLElement
*****\n";
    }
    else {
        echo "***** item is not an instance of SimpleXMLElement
*****\n";
    }
}
?>
```

SimpleXML: Example 2

- Loading an RSS feed and printing the individual news elements:

```
<?php
    $articles = array();
    $rawFeed = file_get_contents('http://feeds.reuters.com/reuters/businessNews');
    $xml = new SimpleXmlElement($rawFeed);
    foreach ($xml->channel->item as $item) {
        $itemRSS = array (
            'source' => 'Reuters',
            'title' => $item->title,
            'desc' => (string) trim($item->description),
            'link' => $item->link,
            'date' => $item->pubDate);
        array_push($articles, $itemRSS);
    }
    if (!empty($articles)) {
        foreach ($articles as $entry) {
            $atomDate = $entry['date'];
            $dateTime = strtotime(substr($atomDate, 5, 11));
            $formattedDate = date('F j, Y', $dateTime);
            echo "<li>" . $formattedDate . " (" . $entry['source'] . ") <a href=\"" . $entry['link'] .
"\n" target=\"_blank\">" . $entry['title'] . "</a></li>\n";
        }
    }
?>
```

SimpleXML: Example 3

□ Using the Twitter Search API

```
<?php
$xml =
simplexml_load_file('http://search.twitter.com/search.atom?q=watson');
foreach ($xml->entry as $item) {
    echo "AUTHOR => " . $item->author->name . "\n";
    echo "AUTHOR'S URL=> " . $item->author->uri .
"\n";
    echo "AUTHOR'S AVATAR => " . $item-
>link[1]['href'] . "\n";
    echo "CONTENT => $item->content\n";
    echo "UPDATED => $item->updated\n\n";
}
?>
```

Variable Scoping: Three Scopes

- Local - Default; only in this page or function
- Global - Makes page variables available in functions. Unlike in C/C++ where using global variables is considered a cardinal sin, globals are used quite frequently in PHP
- Static - Local to function, doesn't change between invocations

Variable Scoping: Example

```
<?php
$a = "one";
function var1()
{
    if (isset($a)) {
        echo 'in var1, $a is ' . $a . '<br/>';
    }
    else {
        echo 'in var1, $a isn\'t set!<br/>';
    }
}

function var2()
{
    global $a;
    if (isset($a)) {
        echo 'in var2, $a is ' . $a . '<br/>';
    }
    else {
        echo 'in var2, $a isn\'t set!<br/>';
    }
}
var1();
var2();
?>
```

Result:

```
in var1, $a isn't set!
in var2, $a is one
```

Static Variables



- Inside subroutines, normally variables start out unset
- Static variables retain their value between subroutine calls

Static Variables: Example

```
<?php
function repeat1()
{
    if (!isset($a)) {
        $a = 0;
    }

    $a++;
    echo 'in repeat1, $a is ' . $a . "<br/>";
}

function repeat2()
{
    static $a;
    if (!isset($a)) {
        $a = 0;
    }

    $a++;
    echo 'in repeat2, $a is ' . $a . "<br/>";
}

for ($i = 0; $i < 5; $i++) {
    repeat1();
}

for ($i = 0; $i < 5; $i++) {
    repeat2();
}

?>
```

Result:

```
in repeat1, $a is 1
in repeat1, $a is 1
in repeat1, $a is 1
in repeat1, $a is 1
in repeat1, $a is 1
in repeat2, $a is 1
in repeat2, $a is 2
in repeat2, $a is 3
in repeat2, $a is 4
in repeat2, $a is 5
```

Call by Value and Reference

- Call by value - Default; changes to a subroutine's parameter don't matter
- Call by reference - Allows a subroutine to change variables in the outside world; add a preceding & to a subroutine variable; same as in C/C++

Call by Value and Reference

```
<?php
    $b = "unchanged";
    $c = "unchanged";

    function var3($a)
    {
        $a = 'hoho';
    }

    function var4(&$a)
    {
        $a = 'haha';
    }

    echo 'before var3 $b is ' . $b . "<br/>";
    var3($b);
    echo 'after var3 $b is ' . $b . "<br/>";
    echo 'before var4 $c is ' . $c . "<br/>";
    var4($c);
    echo 'after var4 $c is ' . $c . "<br/>";
?>
```

Result:

```
before var3 $b is unchanged
after var3 $b is unchanged
before var4 $c is unchanged
after var4 $c is haha
```

PHP Headers

- A way for you to control the HTTP header
- By default, a PHP script's output is HTML and its result code is 200
- Use header function to render content to different format (e.g., PDF, PNG), result code, or even location `urlencode`
- Examples:
 - `header("Content-type: application/xml");`
 - `header("HTTP/1.1 500 Server Error");`
 - `header("Content-type: image/jpg");`
`echo file_get_contents('facepalm.jpg');`
 - `header("Location: http://www.google.com/");`

CAPTCHAs

