PHP Control Structures and Arrays

Lecture 18

Robb T. Koether

Hampden-Sydney College

Mon, Feb 27, 2012
- PHP Decision Structures
  - if Statements
  - switch Statements

- PHP Iteration Structures
  - while Statements
  - for Statements

- PHP Arrays

- Associative Arrays

- MySQL and Associative Arrays
Outline

1. PHP Decision Structures
   - if Statements
   - switch Statements

2. PHP Iteration Structures
   - while Statements
   - for Statements

3. PHP Arrays

4. Associative Arrays

5. MySQL and Associative Arrays
if Statements

The if Statement

```php
if (condition)
    true-block;
else
    false-block;
```

- if statements in PHP are just like if statements in C.
- The else part is optional.
- If the block contains more than one statement, then group them using curly braces.
Outline

1. PHP Decision Structures
   - `if` Statements
   - `switch` Statements

2. PHP Iteration Structures
   - `while` Statements
   - `for` Statements

3. PHP Arrays

4. Associative Arrays

5. MySQL and Associative Arrays
switch Statements

The switch Statement

```
switch (expression)
{
    case value:
        case-block;
        break;
    :
    default:
        default-block;
        break;
}
```

- switch statements in PHP are just like switch statements in C.
Outline

1. PHP Decision Structures
   - if Statements
   - switch Statements

2. PHP Iteration Structures
   - while Statements
   - for Statements

3. PHP Arrays

4. Associative Arrays

5. MySQL and Associative Arrays
Outline

1. PHP Decision Structures
   - if Statements
   - switch Statements

2. PHP Iteration Structures
   - while Statements
   - for Statements

3. PHP Arrays

4. Associative Arrays

5. MySQL and Associative Arrays
while Statements

The while Statement

while (condition)  
  while-block;

- while statements in PHP are just like while statements in C.
Outline

1. PHP Decision Structures
   - if Statements
   - switch Statements

2. PHP Iteration Structures
   - while Statements
   - for Statements

3. PHP Arrays

4. Associative Arrays

5. MySQL and Associative Arrays
The for Statement

for (initial; condition; update)
for-block;

- for statements in PHP are just like for statements in C.
Outline

1. PHP Decision Structures
   - if Statements
   - switch Statements

2. PHP Iteration Structures
   - while Statements
   - for Statements

3. PHP Arrays

4. Associative Arrays

5. MySQL and Associative Arrays
### Initializing an Array

```php
$a = array(5, 10, 15, 20);
$b[0] = 5;
$b[1] = 10;
$b[2] = 15;
$b[3] = 20;
```

- Arrays may be initialized by using the keyword `array`.
- Or, the array elements may be assigned individually without using the keyword.
Displaying Arrays

Displaying an Array

```php
$a = array(5, 10, 15, 20);
for ($i = 0, $i < count($a), $i++)
    echo $a[$i] . ' ';```

- To display an array, we may use a `for` loop.
- The above loop will produce
  
  5 10 15 20
Displaying Arrays

The `foreach` Statement

```
foreach ($a as $b)
  echo $b . ' ';  
```

- Or, we may use a `foreach` loop.
The \texttt{print\_r()} Function

\texttt{print\_r($a);}

- Or, we may use the \texttt{print\_r} function.
- This is more appropriate for debugging.
- The above statement will produce the following.

The `var_dump()` Function

```php
var_dump($a);
```

- Or, we may use the `var_dump` function.
- This is very helpful in debugging.
- The above statement will produce the following.

```plaintext
array(4) { 
  [0] => int(5) 
  [1] => int(10) 
  [2] => int(15) 
  [3] => int(20) 
}
Heterogeneous Arrays

```php
$a = array(5, "dog", true, 6.99);
var_dump($a);
```

- Arrays in PHP may be heterogeneous.
In PHP, arrays may be **associative** instead of **numerically indexed**.

In an associative array, two lists of values are associated as a list of pairs.

- The first member of each pair serves as the index.
- The second member of each pair serves as the value.

Each index must be unique.
$author = array(
    "A Tale of Two Cities" => "Charles Dickens",
    "Treasure Island" => "Robert Louis Stevenson",
    "Tom Sawyer" => "Mark Twain",
    "Oliver Twist" => "Charles Dickens",
    "Roughing It" => "Mark Twain"
);
Associative Arrays

$title = "Treasure Island";
echo "The book "$title" was written by " . $author[$title];

- The above code will produce the following.

The book Treasure Island was written by Robert Louis Stevenson
Associative Arrays and `foreach`

```php
foreach ($author as $a)
    echo "The author is " . $a . '<br/>';
```

- The above code will produce the following.

  The author is Charles Dickens
  The author is Robert Louis Stevenson
  The author is Mark Twain
  The author is Charles Dickens
  The author is Mark Twain
Associative Arrays and **foreach**

```php
foreach ($author as $t => $a)
    echo $a . " wrote " . $t . '<br/>';
```

- The above code will produce

  Charles Dickens wrote A Tale of Two Cities
  Robert Louis Stevenson wrote Treasure Island
  Mark Twain wrote Tom Sawyer
  Charles Dickens wrote Oliver Twist
  Mark Twain wrote Roughing It
echo "<table>";
echo "<tr><th>Title</th><th>Author</th></tr>";
foreach ($author as $t => $a)
{
    echo "<tr><td>" . $t . "</td>";
    echo "<td>" . $a . "</td></tr>";
}
echo "</table>";

- The above code will produce a table of titles and authors.
Outline

1. PHP Decision Structures
   - if Statements
   - switch Statements

2. PHP Iteration Structures
   - while Statements
   - for Statements

3. PHP Arrays

4. Associative Arrays

5. MySQL and Associative Arrays
As we will soon see, MySQL returns tuples as associative arrays. The attribute name is the index. The attribute value is the value.