Slide 3.10: PHP arrays (cont.)
Slide 3.12: PHP looping (cont.)
Home

PHP Looping

Looping statements in PHP are used to execute the same block of code a specified number of times. PHP has the following looping statements:

- **`while`**—loops through a block of code if and as long as a specified condition is true.
- **`do...while`**—loops through a block of code once, and then repeats the loop as long as a special condition is true.
- **`for`**—loops through a block of code a specified number of times.
- **`foreach`**—loops through a block of code for each element in an array.

The while Statement

The while statement will execute a block of code if and as long as a condition is true.

```
while ( condition )
  code to be executed;
```

The example prints the random number between 0 and 9 until the random number matches the input value:

- Compare the web input to character digits instead of integer digits.
- `TRUE` and `FALSE` are a boolean type of PHP.
- The `rand(min,max)` function returns a random number between `min` and `max`, inclusively.

```
<html><body>
<?php
 $input =         ;
 if ( ( $input < '     ' ) ||
      ( $input > '     ' ) )
  echo "$input is not a digit!";
 else {
  while ( true ) {
   $rand = rand( 0, 9 );
   if ( $input == $rand ) {
    echo "Found the random $rand.";
    break;
   }
   else
    echo "The random is $rand.";
  }
```

- `break` ends execution of the current `for`, `foreach`, `while`, `do-while` or `switch` structure.

```
 }
?>
</body></html>
```

Check Source Reset

Slide 3.10: PHP arrays (cont.)
Slide 3.12: PHP looping (cont.)
Home

PRINT VERSION