Programming PHP

Santosh Kalwar, 24.03.2022

PHP Topics

General Discussions

PHP Basic Operations
- Control Structures
- if statement
- if-elseif-else
- ternary operators
- switch
- while loop
- for loop
- foreach-loop
- include and require files

- Practice / Classroom coding

PHP Basic Operations - The if Statement

The *if* statement executes some code if one condition is true.

Syntax

```
if (condition) {
  code to be executed if condition is true;
}
```

```php
<?php
$t = date("H");

if ($t < "20") {
  echo "Have a good day!";
}
?>
```

PHP Basic Operations - The if...else Statement

The `if...else` statement executes some code if a condition is true and another code if that condition is false.

Syntax

```
if (condition) {
  code to be executed if condition is true;
} else {
  code to be executed if condition is false;
}
```

```php
<?php
$t = date("H");

if ($t < "20") {
  echo "Have a good day!";
} else {
  echo "Have a good night!";
}
?>
```

PHP Basic Operations - The if...elseif...else Statement

The `if...elseif...else` statement executes different codes for more than two conditions.

Syntax

```
if (condition) {
  code to be executed if this condition is true;
} elseif (condition) {
  code to be executed if first condition is false and this condition is true;
} else {
  code to be executed if all conditions are false;
}
```

```php
<?php
$t = date("H");

if ($t < "10") {
  echo "Have a good morning!";
} elseif ($t < "20") {
  echo "Have a good day!";
} else {
  echo "Have a good night!";
}
?>
```

PHP Basic Operations - ternary operator

Ternary operator is shorthand for `if...else` statement.

Condition ? Task A : Task B

The ternary operator first checks the condition on its left. If it is true, it performs task A. Else, it performs task B.

Syntax

```php
<?php

if (condition) {
    $result = value1;
} else {
    $result = value2;
}
```

You can use this:

```php
$result = condition ? value1 : value2;
```

PHP Basic Operations - ternary operator

```php
<?php

$is_user_logged_in = false;

if ($is_user_logged_in) {
  $title = 'Logout';
} else {
  $title = 'Login';
}
```

In this example, the $title will be 'Login' because the $is_user_logged_in is set to false. The code is quite lengthy. And you can make it shorter by using the ternary operator as follows:

```php
<?php

$is_user_logged_in = false;

$title = $is_user_logged_in ? 'Logout' : 'Login';
```

PHP Basic Operations - switch Statement

The `switch` statement is used to perform different actions based on different conditions.

Use the `switch` statement to **select one of many blocks of code to be executed**.

Syntax

```
switch (n) {
  case label1:
    code to be executed if n=label1;
    break;
  case label2:
    code to be executed if n=label2;
    break;
  case label3:
    code to be executed if n=label3;
    break;
    ...
  default:
    code to be executed if n is different from all labels;
}
```

PHP Basic Operations - switch Statement

```php
<?php
$favcolor = "red";

switch ($favcolor) {
  case "red":
    echo "Your favorite color is red!";
    break;
  case "blue":
    echo "Your favorite color is blue!";
    break;
  case "green":
    echo "Your favorite color is green!";
    break;
  default:
    echo "Your favorite color is neither red, blue, nor green!";
}
?>
```

PHP Basic Operations - while Loop

The `while` loop - Loops through a block of code as long as the specified condition is true.

The `while` loop executes a block of code as long as the specified condition is true.

Syntax

```
while (condition is true) {
  code to be executed;
}
```

Examples

The example below displays the numbers from 1 to 5:

```php
<?php
$x = 1;

while($x <= 5) {
  echo "The number is: $x <br>";
  $x++;
}
?>
```

PHP Basic Operations - for Loop

The `for` loop - Loops through a block of code a specified number of times. In other words, the `for` loop is used when you know in advance how many times the script should run.

Syntax

```
for (init counter; test counter; increment counter) {
  code to be executed for each iteration;
}
```

Parameters:

- *init counter*: Initialize the loop counter value
- *test counter*: Evaluated for each loop iteration. If it evaluates to TRUE, the loop continues. If it evaluates to FALSE, the loop ends.
- *increment counter*: Increases the loop counter value

Examples

The example below displays the numbers from 0 to 10:

```php
<?php
for ($x = 0; $x <= 10; $x++) {
  echo "The number is: $x <br>";
}
?>
```

PHP Basic Operations - foreach Loop

The `foreach` loop works only on arrays, and is used to loop through each key/value pair in an array.

Syntax

```
foreach ($array as $value) {
  code to be executed;
}
```

For every loop iteration, the value of the current array element is assigned to $value and the array pointer is moved by one, until it reaches the last array element.

Examples

The following example will output the values of the given array ($colors):

```php
<?php
$colors = array("red", "green", "blue", "yellow");

foreach ($colors as $value) {
  echo "$value <br>";
}
?>
```

PHP Basic Operations - Include/Require Files

It is possible to insert the content of one PHP file into another PHP file (before the server executes it), with the include or require statement.

The include and require statements are identical, except upon failure:

- `require` will produce a fatal error (E_COMPILE_ERROR) and stop the script
- `include` will only produce a warning (E_WARNING) and the script will continue

Syntax

```
include 'filename';

or

require 'filename';
```

Example

Assume we have a standard footer file called "footer.php", that looks like this:

```php
<?php
echo "<p>Copyright &copy; 2022-" . date("Y") . " bc.fi</p>";
?>
```

To include the footer file in a page, use the `include` statement:

```html
<html>
<body>

<h1>Welcome to my home page!</h1>
<p>Some text.</p>
<p>Some more text.</p>
<?php include 'footer.php';?>

</body>
</html>
```

Practice / Classroom Coding

- **PHP_practice01/3.php**:
  - Make an if statement with elseif and else to finally display string saying, I love PHP
  - Make a forloop that displays 10 numbers
  - Make a switch statement that test against one condition with 5 cases

- **PHP_practice03/section_a/c02/if-statement.php**

  - Create a simple if statement to greet user if the name is not empty.

- **PHP_practice03/section_a/c02/if-else-statement.php**

  - Create a simple if else statement to check if candy is in the stock or no. If candy is not in stock then print message "Sold Out", if it is availaible in stock, print message "In Stock"

- **PHP_practice03/section_a/c02/ternary-operator.php**

  - Re-factor using ternary operator: Create a simple if else statement to check if candy is in the stock or no. If candy is not in stock then print message "Sold Out", if it is availaible in stock, print message "In Stock"

Practice / Classroom Coding

- **PHP_practice03/section_a/c02/ternary-operator.php**

  Re-factor using ternary operator: Create a simple if else statement to check if candy is in the stock or no. If candy is not in stock then print message "Sold Out", if it is availaible in stock, print message "In Stock"

- **PHP_practice03/section_a/c02/if-else-if-statement.php**

  Create a simple if else if statement to check if candy is in the stock or is coming soon or sold out. If candy is not in stock then print message "Sold Out", if it is availaible in stock, print message "In Stock"

- **PHP_practice03/section_a/c02/switch-statement.php**

  Create a simple switch statement to get 20% off chocolates on Monday, 20% off mints on Tuesday and in all other cases it should show "Buy three packs, get one free"

- **PHP_practice03/section_a/c02/while-loop.php**

  Create a simple while loop to find prices for multiple packs of candy. For example if one pack costs $1.99 how much would 5 pack costs. Display the prices for all the 5 packs of candy.

Practice / Classroom Coding

- **PHP_practice03/section_a/c02/for-loop.php**

  Create a simple for loop to find prices of multiple packs of candy. Let us assume one pack of candy costs $1.99. How much 10 packs cost? Display each packs costs in the web page.

  - **PHP_practice03/section_a/c02/for-loop-higher-counter.php**

  Create a simple for loop to find prices of multiple higher packs of candy. Let us assume customer wants from 10 packs to 100 packs of candies. How much 10 to 100 packs cost? Display from 10 packs to 100 packs cost in the web page.

  - **PHP_practice03/section_a/c02/foreach-loop.php**

  Create a simple foreach loop for candy products e.g. Toffee costs 2.99, Mints cost 1.99 and Fudge costs 3.49 and show it in the web page table.

  - **PHP_practice03/section_a/c02/include-and-require-files.php**

  Write a PHP Code to include header.php, footer.php and check the stock of candy. Let us assume you have 25 stock of candy so check if you have "Good availability" or you have "low stock" or if you are running "Out of stock"