

PHP Introduction



SOFTWARE DEVELOPMENT WORKSHOP I

What is PHP?



- PHP is an acronym for "PHP: Hypertext Preprocessor"
- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server
- PHP is free to download and use
 - It is powerful enough to be at the core of the biggest blogging system on the web (WordPress)!
 - It is deep enough to run the largest social network (Facebook)!
 - It is also easy enough to be a beginner's first server side language!

What Can PHP Do?



- PHP can generate **dynamic page** content
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect **form data**
- PHP can send and receive **cookies**
- PHP can add, delete, modify data in your **database**
- PHP can be used to **control user-access**
- PHP can **encrypt data**



- Why PHP?
- PHP runs on various **platforms** (Windows, Linux, Unix, Mac OS X, etc.)
- PHP is **compatible** with almost all servers used today (Apache, IIS, etc.)
- PHP **supports** a wide range of databases
- PHP is **free**. Download it from the official PHP resource: www.php.net
- PHP is **easy to learn** and runs efficiently on the **server side**

Basic PHP Syntax



- A PHP script can be placed anywhere in the document.
- A PHP script starts with `<?php` and ends with `?>`:

```
<?php
// PHP code goes here
?>
```

- The default file extension for PHP files is `".php"`.
- We'll be using PHP statements inside HTML files as well

Basic PHP Syntax



- Comments in PHP

```
<?php
// This is a single-line comment

# This is also a single-line comment

/*
This is a multiple-lines comment block
that spans over multiple
lines
*/

// You can also use comments to leave out parts of a code line
$x = 5 /* + 15 */ + 5;
echo $x;
?>
```

Basic PHP Syntax



- PHP Case Sensitivity - variables

```
<?php
$color = "red";
echo "My car is " . $color . "<br>";
echo "My house is " . $COLOR . "<br>";
echo "My boat is " . $coLOR . "<br>";
?>
```

My car is red
My house is
My boat is

- PHP **(not)** Case Sensitivity - keywords

```
<?php
ECHO "Hello World!<br>";
echo "Hello World!<br>";
EcHo "Hello World!<br>";
?>
```

Hello World!
Hello World!
Hello World!

Variables



- Creating (Declaring) PHP Variables

```
<?php
$txt = "Hello world!";
$x = 5;
$y = 10.5;

echo $txt;
echo "<br>";
echo $x;
echo "<br>";
echo $y;
?>
```

```
Hello world!
5
10.5
```

Note: When you assign a text value to a variable, put quotes around the value.

Note: Unlike other programming languages, PHP has no command for declaring a variable. It is created the moment you first assign a value to it.

Variables



A variable can have a short name (like `$x` and `$y`) or a more descriptive name (`$age`, `$carname`, `$total_volume`).

Rules for PHP variables:

- A variable starts with the **\$** sign, followed by the name of the variable
- A variable name must start with a **letter** or the **underscore** character
- A variable name **cannot** start with a **number**
- A variable name can only contain **alpha-numeric** characters and **underscores** (A-z, 0-9, and `_`)
- Variable names are case-sensitive (`$age` and `$AGE` are two different variables)

Variables



Incorrect

\$my Name

4name!

\$*name

Correct

\$my_Name

\$_23fine

\$myName

Output Variables



- The PHP **echo** statement is often used to output data to the screen.

```
$txt = "W3Schools.com";  
echo "I love $txt!";
```

I love W3Schools.com!

```
$txt = "W3Schools.com";  
echo "I love " . $txt . "!";
```

I love W3Schools.com!

```
$x = 5;  
$y = 4;  
echo $x + $y; // 9
```

sum of two variables:

Variables Scope



- In PHP, variables can be declared anywhere in the script.
- The scope of a variable is the part of the script where the variable can be referenced/used.
- PHP has three different variable scopes:
 - local
 - global
 - static

Global Scope



- variables declared **outside** a function can only be accessed outside a function:

```
<?php
$x = 5; // global scope

function myTest() {
    // using x inside this function will generate an error
    echo "<p>Variable x inside function is: $x</p>";
}
myTest();

echo "<p>Variable x outside function is: $x</p>";
?>
```

Variable x inside function is:

Variable x outside function is: 5

LOCAL SCOPE



- variables declared within a function can only be accessed within that function:

```
<?php
function myTest() {
    $x = 5; // local scope
    echo "<p>Variable x inside function is: $x</p>";
}
myTest();

// using x outside the function will generate an error
echo "<p>Variable x outside function is: $x</p>";
?>
```

You can have local variables with the same name in different functions, because local variables are only recognized by the function in which they are declared.

Variable x inside function is: 5

Variable x outside function is:

The global Keyword



- The global keyword is used to access a global variable from within a function.

```
$x = 5;  
$y = 10;  
function myTest() {  
    global $x, $y;  
    $y = $x + $y;  
}  
myTest(); // run function  
echo $y; // output the new value for variable $y
```

15

The static Keyword



- Normally, when a function is completed/executed, all of its variables are deleted.
- sometimes we want a local variable NOT to be deleted
- use the **static** keyword when you first declare the variable:

```
function myTest() {  
    static $x = 0;  
    echo $x;  
    $x++;  
}  
myTest();  
echo "<br>";  
myTest();  
echo "<br>";  
myTest();
```

0
1
2

Each time the function is called, that variable will still have the information it contained from the last time the function was called.

echo and print Statements



- In PHP there are two basic ways to get output: echo and print.
- In SDW1, we'll mainly use the echo statement
- echo and print are more or less the same.
 - They are both used to output data to the screen.

echo Statement



- The echo statement can be used with or without parentheses:
 echo or echo().
- Display Text
- Display Variables

echo Statement



- Display Text

```
<?php
echo "<h2>PHP is Fun!</h2>";
echo "Hello world!<br>";
echo "I'm about to learn PHP!<br>";
echo "This ", "string ", "was ", "made
", "with multiple parameters.";
?>
```

PHP is Fun!

Hello world!

I'm about to learn PHP!

This string was made with multiple parameters.

echo Statement



- Display Variables

```
<?php
$txt1 = "Learn PHP";
$txt2 = "W3Schools.com";
$x = 5;
$y = 4;

echo "<h2>" . $txt1 . "</h2>";
echo "Study PHP at " . $txt2
. "<br>";
echo $x + $y;
?>
```

Learn PHP

Study PHP at W3Schools.com

Download Task



- Download task from iSpace
- PHP has similar statements/commands to JavaScript
- Using the website:
<https://www.quanzhanketang.com/php/default.html>
- Complete the task and upload to the exercise section
- If you are not sure about how to do one of the task, please ask – we are here to help