PHP 5 Introduction

What You Should Already Know

you should have a basic understanding of the following:

- HTML
- CSS

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

What is a PHP File?

- PHP files can contain text, HTML, CSS, JavaScript, and PHP code
- PHP code are executed on the server, and the result is returned to the browser as plain HTML
- PHP files have extension ".php"

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

With PHP you are not limited to output HTML. You can output images, PDF files, and even Flash movies. You can also output any text, such as XHTML and XML.

Why PHP?

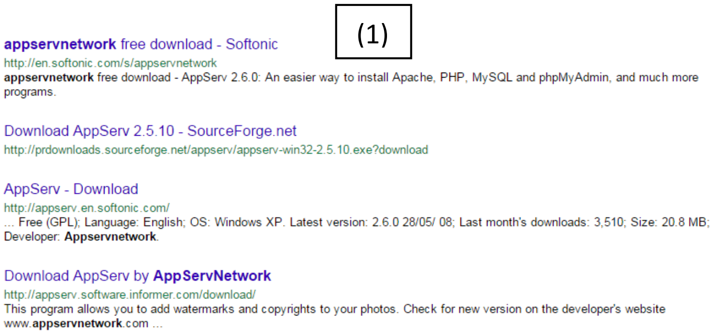
- PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)

- 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

PHP 5 Installation

To start using PHP, you can:

- Find a web host with PHP and MySQL support
- Install a web server on your own PC, and then install PHP and MySQL



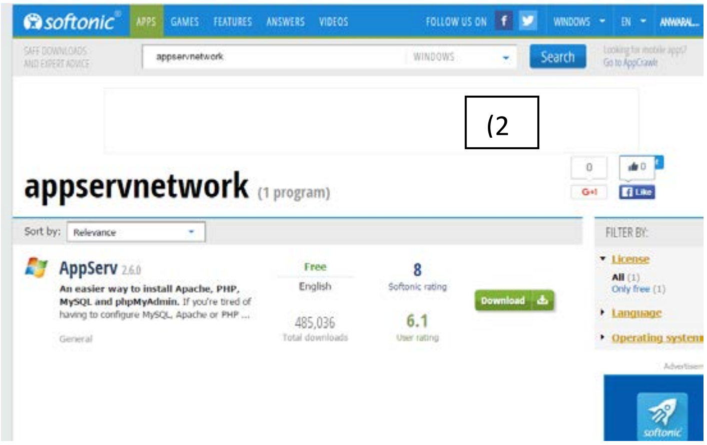
(1)

appservnetwork free download - Softonic
http://en.softonic.com/s/appservnetwork
appservnetwork free download - AppServ 2.6.0: An easier way to install Apache, PHP, MySQL and phpMyAdmin, and much more programs.

Download AppServ 2.5.10 - SourceForge.net
http://prdownloads.sourceforge.net/appserv/appserv-win32-2.5.10.exe?download

AppServ - Download
http://appserv.en.softonic.com/
... Free (GPL); Language: English; OS: Windows XP. Latest version: 2.6.0 28/05/ 08; Last month's downloads: 3,510; Size: 20.8 MB; Developer: **Appservnetwork**.

Download AppServ by **AppServNetwork**
http://appserv.software.informer.com/download/
This program allows you to add watermarks and copyrights to your photos. Check for new version on the developer's website www.**appservnetwork**.com ...



(2

appservnetwork (1 program)

Search: appservnetwork — WINDOWS

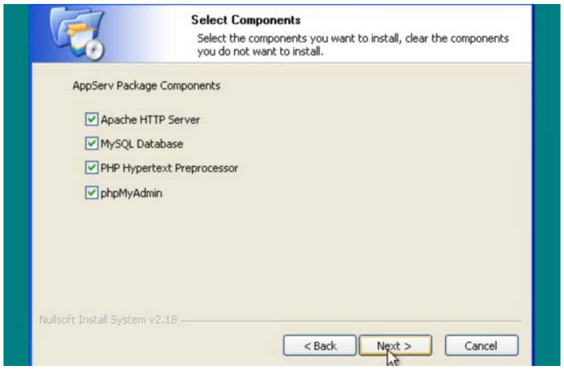
Sort by: Relevance

AppServ 2.6.0 — An easier way to install Apache, PHP, MySQL and phpMyAdmin. If you're tired of having to configure MySQL, Apache or PHP ... — General

Free — English — 485,036 Total downloads — 8 Softonic rating — 6.1 User rating

License: All (1), Only free (1)

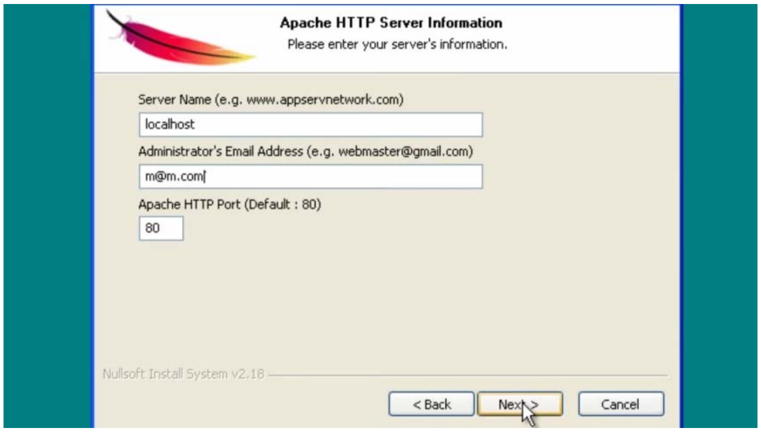
(3) Click Download.



Select Components
Select the components you want to install, clear the components you do not want to install.

AppServ Package Components

☑ Apache HTTP Server
☑ MySQL Database
☑ PHP Hypertext Preprocessor
☑ phpMyAdmin



Apache HTTP Server Information

Please enter your server's information.

Server Name (e.g. www.appservnetwork.com)

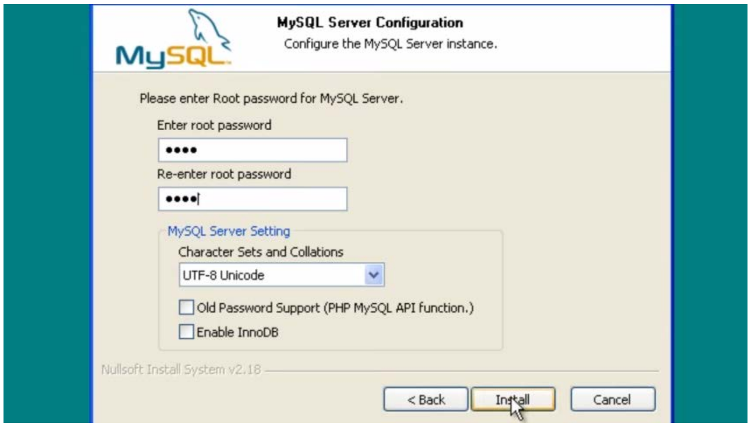
localhost

Administrator's Email Address (e.g. webmaster@gmail.com)

m@m.com

Apache HTTP Port (Default : 80)

80



MySQL Server Configuration

Configure the MySQL Server instance.

Please enter Root password for MySQL Server.

Enter root password

••••

Re-enter root password

••••

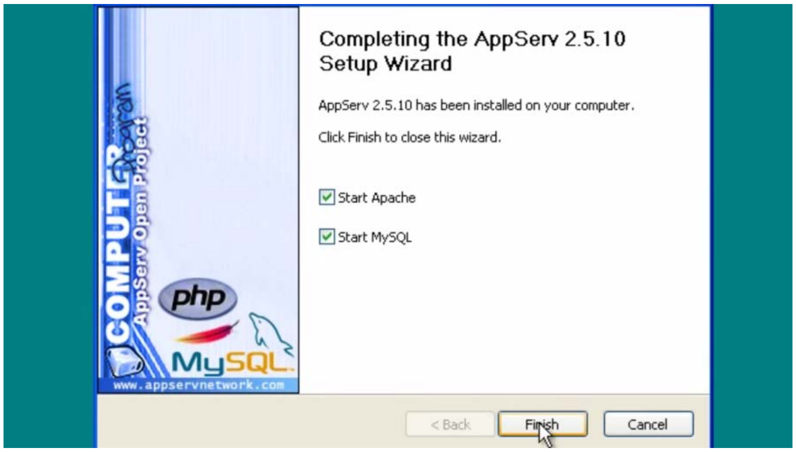
MySQL Server Setting

Character Sets and Collations

UTF-8 Unicode

□ Old Password Support (PHP MySQL API function.)

□ Enable InnoDB



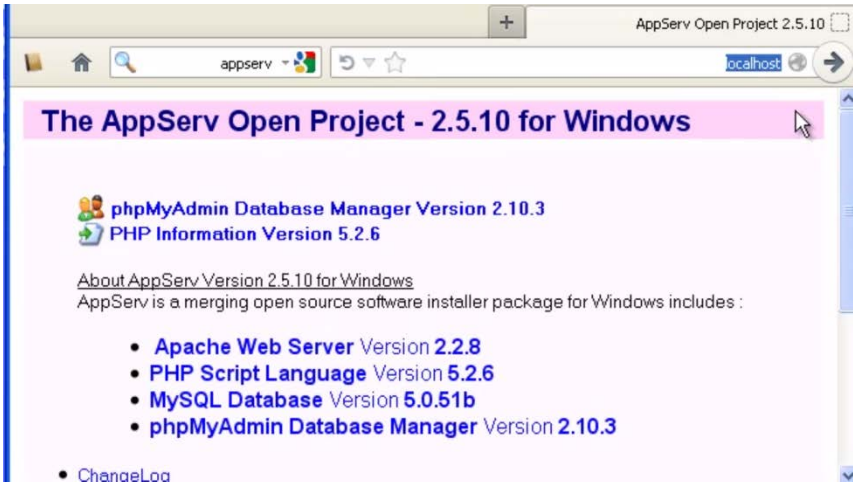
Completing the AppServ 2.5.10 Setup Wizard

AppServ 2.5.10 has been installed on your computer.

Click Finish to close this wizard.

☑ Start Apache

☑ Start MySQL



The AppServ Open Project - 2.5.10 for Windows

phpMyAdmin Database Manager Version 2.10.3

PHP Information Version 5.2.6

About AppServ Version 2.5.10 for Windows

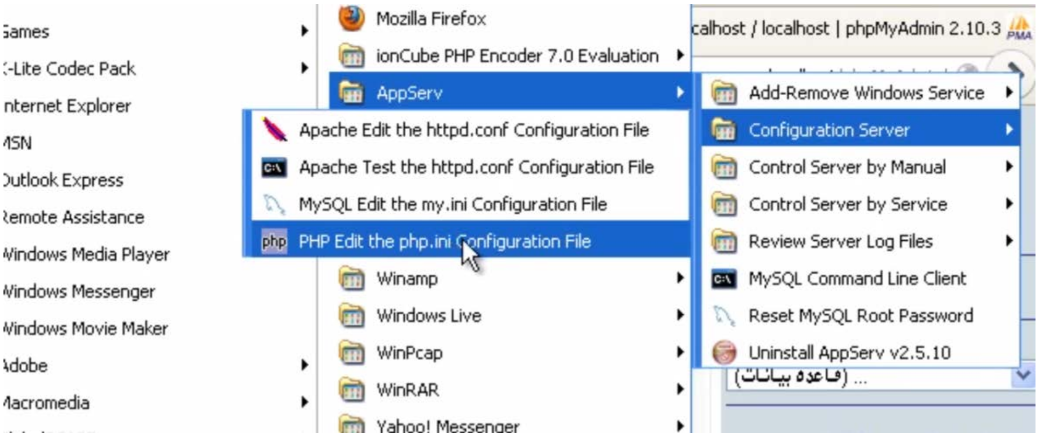
AppServ is a merging open source software installer package for Windows includes :

- Apache Web Server Version 2.2.8
- PHP Script Language Version 5.2.6
- MySQL Database Version 5.0.51b
- phpMyAdmin Database Manager Version 2.10.3

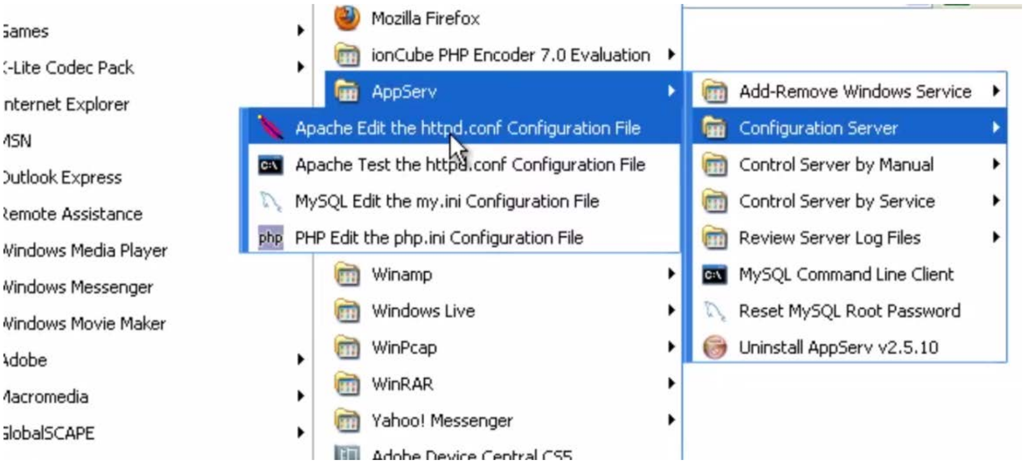
- ChangeLog

To settings the local server on pc:

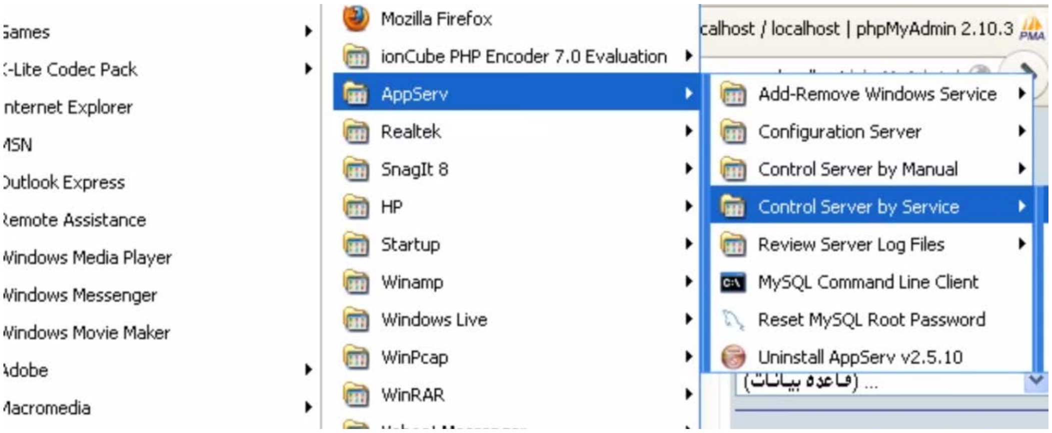
1) Start → All Programs → AppServ → Configuration Server → PHP Edit the php.ini Configuration File → find → max_execution time (Set to 120 seconds) →Save.


Games
K-Lite Codec Pack
Internet Explorer
MSN
Outlook Express
Remote Assistance
Windows Media Player
Windows Messenger
Windows Movie Maker
Adobe
Macromedia
Mozilla Firefox
ionCube PHP Encoder 7.0 Evaluation
AppServ
Apache Edit the httpd.conf Configuration File
Apache Test the httpd.conf Configuration File
MySQL Edit the my.ini Configuration File
PHP Edit the php.ini Configuration File
Winamp
Windows Live
WinPcap
WinRAR
Yahoo! Messenger
calhost / localhost | phpMyAdmin 2.10.3
Add-Remove Windows Service
Configuration Server
Control Server by Manual
Control Server by Service
Review Server Log Files
MySQL Command Line Client
Reset MySQL Root Password
Uninstall AppServ v2.5.10
(قاعدة بيانات) ...

2) Start → All Programs → AppServ → Configuration Server → Apache Edit the httpd.conf Configuration File → find → rewrite_module → Delete # →Save.


Games
K-Lite Codec Pack
Internet Explorer
MSN
Outlook Express
Remote Assistance
Windows Media Player
Windows Messenger
Windows Movie Maker
Adobe
Macromedia
GlobalSCAPE
Mozilla Firefox
ionCube PHP Encoder 7.0 Evaluation
AppServ
Apache Edit the httpd.conf Configuration File
Apache Test the httpd.conf Configuration File
MySQL Edit the my.ini Configuration File
PHP Edit the php.ini Configuration File
Winamp
Windows Live
WinPcap
WinRAR
Yahoo! Messenger
Adobe Device Central CS5
Add-Remove Windows Service
Configuration Server
Control Server by Manual
Control Server by Service
Review Server Log Files
MySQL Command Line Client
Reset MySQL Root Password
Uninstall AppServ v2.5.10

3) Start → All Programs → AppServ → Control Server by Service → Apache Restart (to restart the server).



PHP 5 Syntax

A PHP script is executed on the server, and the plain HTML result is sent back to the browser.

Basic PHP Syntax

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

```
<?php
echo "Hello World!";
?>
```

The default file extension for PHP files is ".php". A PHP file normally contains HTML tags, and some PHP scripting code.

Comments in PHP

A comment in PHP code is a line that is not read/executed as part of the program. Its only purpose is to be read by someone who is looking at the code. Comments can be used to:

- Let others understand what you are doing
- Remind yourself of what you did - Most programmers have experienced coming back to their own work a year or two later and having to re-figure out what they did. Comments can remind you of what you were thinking when you wrote the code

PHP supports several ways of commenting:

Example

```
<?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;
?>
```

PHP Case Sensitivity

In PHP, all keywords (e.g. if, else, while, echo, etc.), classes, functions, and user-defined functions are NOT case-sensitive. In the example below, all three echo statements below are legal (and equal):

Example

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

However; all variable names are case-sensitive.

In the example below, only the first statement will display the value of the $color variable (this is because $color, $COLOR, and $coLOR are treated as three different variables):

Example

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

PHP 5 Variables

Creating (Declaring) PHP Variables

In PHP, a variable starts with the $ sign, followed by the name of the variable:

Example

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

After the execution of the statements above, the variable **$txt** will hold the value **Hello world!**, the variable **$x** will hold the value **5**, and the variable **$y** will hold the value **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.

PHP 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)

Output Variables

The PHP echo statement is often used to output data to the screen. The following example will show how to output text and a variable:

Example

```
<?php
$name = "Ali Hassan";
echo "My  Name is  $name!";
?>
```

The following example will produce the same output as the example above:

Example

```
<?php
$name = "Ali Hassan";
echo " My  Name is  " . $name . "!";
?>
```

The following example will output the sum of two variables:

Example

```
<?php
$x = 5;  $y = 4;  echo $x + $y;
?>
```

PHP 5 echo and print Statements

In PHP there are two basic ways to get output: echo and print. In this tutorial we use echo (and print) in almost every example. So, this chapter contains a little more info about those two output statements.

PHP echo and print Statements

echo and print are more or less the same. They are both used to output data to the screen. The differences are small: echo has no return value while print has a return value of 1 so it can be used in expressions. echo can take multiple parameters (although such usage is rare) while print can take one argument. echo is marginally faster than print.

The PHP echo Statement

The echo statement can be used with or without parentheses: echo or echo().

Display Text

The following example shows how to output text with the echo command (notice that the text can contain HTML markup):

Example

```
<?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.";
?>
```

The PHP print Statement

The print statement can be used with or without parentheses: print or print().

Display Text

The following example shows how to output text with the print command (notice that the text can contain HTML markup):

Example

```
<?php
print "<h2>PHP is Fun!</h2>";
print "Hello world!<br>";
print "I'm about to learn PHP!";
?>
```

PHP 5 Data Types

Variables can store data of different types, and different data types can do different things. PHP supports the following data types:

- String
- Integer
- Float (floating point numbers - also called double)
- Boolean
- Array
- Object
- NULL

PHP String

A string is a sequence of characters, like "Hello world!". A string can be any text inside quotes. You can use single or double quotes:

Example

```
<?php
$x = "Hello world!";
$y = 'Hello world!';
echo $x;
echo "<br>";
echo $y;
?>
```

PHP Integer

An integer data type is a non-decimal number between -2,147,483,648 and 2,147,483,647. Rules for integers:

- An integer must have at least one digit
- An integer must not have a decimal point
- An integer can be either positive or negative
- Integers can be specified in three formats: decimal (10-based), hexadecimal (16-based - prefixed with 0x) or octal (8-based - prefixed with 0)

In the following example $x is an integer. The PHP var_dump() function returns the data type and value:

Example

```
<?php
$x = 5985;
var_dump($x);
?>
```

PHP Float

A float (floating point number) is a number with a decimal point or a number in exponential form. In the following example $x is a float. The PHP var_dump() function returns the data type and value:

Example

```
<?php
$x = 10.365;
var_dump($x);
?>
```

PHP Boolean

A Boolean represents two possible states: TRUE or FALSE.

```
$x = true;
$y = false;
```

Booleans are often used in conditional testing.

PHP Array

An array stores multiple values in one single variable. In the following example $cars is an array. The PHP var_dump() function returns the data type and value:

Example

```
<?php
$cars = array("Volvo","BMW","Toyota");
var_dump($cars);
?>
```

PHP Object

An object is a data type which stores data and information on how to process that data. In PHP, an object must be explicitly declared. First we must declare a class of object. For this, we use the class keyword. A class is a structure that can contain properties and methods:

Example

```
<?php
class Car {
    function Car() {
        $this->model = "VW";
    }
}

// create an object
$herbie = new Car();

// show object properties
echo $herbie->model;
?>
```

PHP NULL Value

Null is a special data type which can have only one value: NULL. A variable of data type NULL is a variable that has no value assigned to it. **Tip:** If a variable is created without a value, it is automatically assigned a value of NULL. Variables can also be emptied by setting the value to NULL.

PHP 5 Strings

A string is a sequence of characters, like "Ali Hassan!".

PHP String Functions

In this chapter we will look at some commonly used functions to manipulate strings.

Get The Length of a String

The PHP strlen() function returns the length of a string. The example below returns the length of the string "Hello world!":

Example

```
<?php
echo strlen("Hello world!"); // outputs 12
?>
```

Count The Number of Words in a String

The PHP str_word_count() function counts the number of words in a string:

Example

```
<?php
echo str_word_count("Hello world!"); // outputs 2
?>
```

Reverse a String

The PHP strrev() function reverses a string:

Example

```
<?php
echo strrev("Hello world!"); // outputs !dlrow olleH
?>
```

Search For a Specific Text Within a String

The PHP strpos() function searches for a specific text within a string. If a match is found, the function returns the character position of the first match. If no match is found, it will return FALSE. The example below searches for the text "world" in the string "Hello world!":

Example

```
<?php
echo strpos("Hello world!", "world"); // outputs 6
?>
```

Tip: The first character position in a string is 0 (not 1).

Replace Text Within a String

The PHP str_replace() function replaces some characters with some other characters in a string. The example below replaces the text "world" with "Dolly":

Example

```
<?php
echo str_replace("world", "Dolly", "Hello world!"); // outputs Hello Dolly!
?>
```

Complete PHP String Reference

For a complete reference of all string functions, go to Appendix A.

PHP 5 Constants

Constants are like variables except that once they are defined they cannot be changed or undefined.

PHP Constants

A constant is an identifier (name) for a simple value. The value cannot be changed during the script. A valid constant name starts with a letter or underscore (no $ sign before the constant name).

Note: Unlike variables, constants are automatically global across the entire script.

Create a PHP Constant

To create a constant, use the define() function.

Syntax

```
define(name, value, case-insensitive)
```

Parameters:

- *name*: Specifies the name of the constant
- *value*: Specifies the value of the constant
- *case-insensitive*: Specifies whether the constant name should be case-insensitive. Default is false

The example below creates a constant with a **case-sensitive** name:

Example

```
<?php
define("NAME", "Ali Hassan!");
echo NAME;
?>
```

The example below creates a constant with a **case-insensitive** name:

Example

```
<?php
define("NAME", "Ali Hassan!", true);
echo name;
?>
```

Constants are Global

Constants are automatically global and can be used across the entire script.

The example below uses a constant inside a function, even if it is defined outside the function:

Example

```
<?php
define("NAME", "Ali Hassan!");

function myTest() {
    echo NAME;
}

myTest();
?>
```