

PHP - Variables

Variable is a symbol or name that stands for a value. Variables are used for storing values such as numeric values, characters, character strings, or memory addresses so that they can be used in any part of the program. a variable does not need to be declared before adding a value to it. PHP automatically converts the variable to the correct data type, depending on its value.

Syntax

```
<?php
    $a = 10;
    echo $a;
?>
```

Naming Conventions for PHP Variables

All variables in PHP start with a \$ 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 in PHP can only contain alpha-numeric characters and underscores (A-z, 0-9, and _).

A variable name cannot contain spaces.

PHP is a Loosely Type Language

In language such as C, C++ and Java the programmer must declare the name and type of the variable before use it. In PHP the type of the variable does not need to be declared before use it, because types are associated with values rather than variables. As a result a variable can change the type of its value as much as we want.

Variable References

PHP also allows you to do some neat things with variables. It allows you to create aliases for variables, and it also allows you to have variables whose name is a variable. A variable reference, or alias, is a variable assigned to refer to the same information as another variable. To assign an alias to a variable

Environment Variables

PHP has a collection of environment variables, which are system defined variables that are accessible from

anywhere inside the PHP code. All of these environment variables are stored by PHP as arrays. Some you can address directly by using the name of the index position as a variable name. Other can only be accessed through their arrays.

| Variable | Descriptions |
|------------------------|---|
| <code>\$_SERVER</code> | Contains information about the server and the HTTP connection. Analogous to the old <code>\$HTTP_SERVER_VARS</code> array (which is still available, but deprecated). |
| <code>\$_COOKIE</code> | Contains any cookie data sent back to the server from the client. Indexed by cookie name. Analogous to the old <code>\$HTTP_COOKIE_VARS</code> array (which is still available, but deprecated). |
| <code>\$_GET</code> | Contains any information sent to the server as a search string as part of the URL. Analogous to the old <code>\$HTTP_GET_VARS</code> array (which is still available, but deprecated). |
| <code>\$_POST</code> | <code>\$_POST</code> Contains any information sent to the server as a POST style posting from a client form. Analogous to the old <code>\$HTTP_POST_VARS</code> array (which is still available, but deprecated). |
| <code>\$_FILE</code> | Contains information about any uploaded files. Analogous to the old <code>\$HTTP_POST_FILES</code> array (which is still available, but deprecated). |
| <code>\$_ENV</code> | Contains information about environmental variables on the server. Analogous to the old <code>\$HTTP_ENV_VARS</code> array (which is still available, but deprecated). |

Complete Example

```
<?php
$a = 10;
$b = 20;

$c = $a + $b;

echo"The Addition of a and b :".$c;

?>
```

Output

```
The Addition of a and b : 30
```