



syntax

Objectives : This chapter will cover the syntax (grammar) of PHP language. To learn a new programming language, we should know the syntax first. After that only we can master the language.

Topics that we're going to discuss are;

- A:. PHP script tag.
- B:. Comment.
- C:. Statements.
- D:. Expressions.
- E:. Keyword/reserved words.
- F:. Variable,
- G:. Data types.
- H:. Lateral value.
- I:. Printing Messages.
- J:. Concatenation.
- K:. Simple debugging technique.

PHP SCRIPT TAG

PHP scripts is a segment of program written in the HTML file. It is embedded inside the HTML code in order to program the HTML page to act accordingly to the dynamic changes set by the programmer.

<?php ?> is the tag to segregate the PHP script from the rest of the HTML tags. Between the '**<?php**' and '**?>**' lies the PHP script.

Example :

```
<html>
<head>
<title>Today's Date</title>
</head>
<body>
Today's Date (according to this Web server) is <br>
<?php
    print("Hello World!!<br>");
    echo( date("d/m/y") );
?>
</body>
</html>
```

} php script inside a HTML page

B:. COMMENTS

- Comment is code that will not be compiled (ignored when the program is compiled).
- It will not effect the code.
- It is a good practice to write the comment.
- Comment is useful the tell others what is the programming doing.
- Helps programmer to memorize what is he doing.

3 types of comments.

1. Line comment.

Symbol : //

The statements written after the double slash will be ignored.

Example:

```
//display variable value
$num1=70;
echo $num1;
```

2. Block comment – to comment statements or programmers' remark for more than one lines of codes.

Symbol : /*.....*/

Example:

```
<?php
/*   Author      : Khirulnizam Abd Rahman
    Email       : khirulnizam@gmail.com
    Website     : http://dhost.info/kerul/webdev/
    Purpose of script
        Setting the connection for database host
*/
    $db=mysql_connect("172.16.12.1 ", "root ", "kuis123");
?>
```

C:. STATEMENTS

Any statements must be stopped by a semicolon ';'. PHP interpreter will generate an error message if the statement is not ended with the semicolon, the compiler will tell you there is syntax error.

Example:

```
$huruf='a';
echo ("Hello $nama, you're entering PHP world.");
```

D:. EXPRESSIONS

Expression is a statement that produces value.

Example:

```
$jumlah = $hasil1 + $hasil2;
if ($jumlah == 0)
while ($n < 10)
```

E.: VARIABLE

- Memory location to hold a value (a number, string, object, array, etc).
- PHP is a **loosely typed** language.
- means that a single variable may contain any type of data, (integer, floating value, string) In the process, the variable changes type: where it used to contain a number, it
- The same variable can hold different data type at different time
- No need to declare
- Must have dollar sign (\$) in front of variable

Eg :

```
$testvariable = "Three";  
$testvariable = 3;
```

Variable Naming Rules

A variable name must start with a dollar sign (\$).

- A variable name can only contain alpha-numeric characters and underscores (a-Z, 0-9, and _)
- A variable name should not contain spaces. If a variable name should be more than one word, it should be separated with underscore (\$my_string), or with capitalization (\$myString)

Examples of valid variable name.

1. \$nom2
2. \$jumlah
3. \$huruf
4. \$_nama
5. \$for
6. \$FLOAT
7. \$duitringgit
8. \$makanApa
9. \$myName
10. \$tiada2_

Examples of invalid identifier.

1. 2nom (begins with numeric)
2. \$jumlah nombor (there are space [special character])
3. \$1234 (all characters are numeric)
4. \$kerul@hotmail (@ is invalid character)
5. \$too.much ('.' is invalid character)

F.: KEYWORD/RESERVED WORD

- Words that are used as a special instruction in the programming language.
- These words have special meaning in PHP. Some of them represent things which look like functions, some look like constants, and so on--but they're not, really: they are language constructs. You cannot use any of the following words as constants, class names, function or method names. Using them as variable names is generally OK, but could lead to confusion.

Table K.1. List of PHP Keywords

And	or	xor	__FILE__	
__LINE__	array()	as	break	case
class	const	continue	declare	default
Die()	do	echo()	else	elseif
empty()	enddeclare	endfor	endforeach	endif
endswitch	endwhile	eval()	exit()	extends
For	foreach	function	global	if
include()	include_once()	isset()	list()	new
print()	require()	require_once()	return()	static
switch	unset()	use	var	while
__FUNCTION__	__CLASS__	__METHOD__		
PHP 5 Only				
interface	implements	exception	final	php_user_filter
protected	abstract	extends	public	private
throw		clone	try	catch
PHP 4 Only				
		cfunction	old_function	this

G.: LATERAL VALUES

- Lateral is the exact/actual value (what you see is what you get).
- The value could be number(integer or real), character, string, or Boolean.

Integer

- Integer is round number.
- For example, 10 is an integer.

```
Example ;
    $nomborBesar = 10;
```

- Use negative sign to for negative number.

```
Example ;
    $num = -10;
```

Float

```
Example :
    $temp = 37.98789;
    $product = 0.0;
```

Characters

- To assign a character value into a *char* variable, the value must be put inside a open and close quote.
Example : 'A', '?', '7'

```
Example :  
$huruf = 'b';  
$grade= 'A';
```

These are the list of hidden character in PHP (similar like in C).

\n	New line
\t	Tab (a tab space)
\b	Backspace
\'	Single quote
\"	Double quote

Boolean

- There are only two values for Boolean variable,
- **true** and
- **false**

```
Example :  
$seatalready = true;  
$isInteger = false;
```

String

- A group of characters.
- Marked with double quote (".....").

```
Example:  
"123434"  
"T"  
"I love PHP"  
"\tPHP tak suka saya."  
"\n I am using PHP4"  
"We need PHP to do web programming."  
"This string will \n be printed \n in 3 lines"
```

H.: PRINTING MESSAGES

```
<?  
$welcome_text = "Hello and welcome to my website. ";  
print $welcome_text;  
?>  
We can also use echo
```

```
<?  
$welcome_text = "Hello and welcome to my website.";  
echo $welcome_text;  
?>
```

I.: CONCATENATION

- Concatenation is to join(or to merge) several strings into one.
- In PHP we usually use "." (dot) to do concatenation.

```
<?
    $name = "Kerul";
    $age=21;
    echo "My name is ".$name. "<br>";
    echo "I\'m ".$age. "years old...<br>";
?>
```

concatenator variable lateral string

Simpler way of doing concatenation...

Used to merge a lateral string and value of a variable.

```
<?
    $name = "Kerul";
    $age=21;
    echo "My name is $name <br>";
    echo "I am $age years old...<br>";
?>
```

Example:

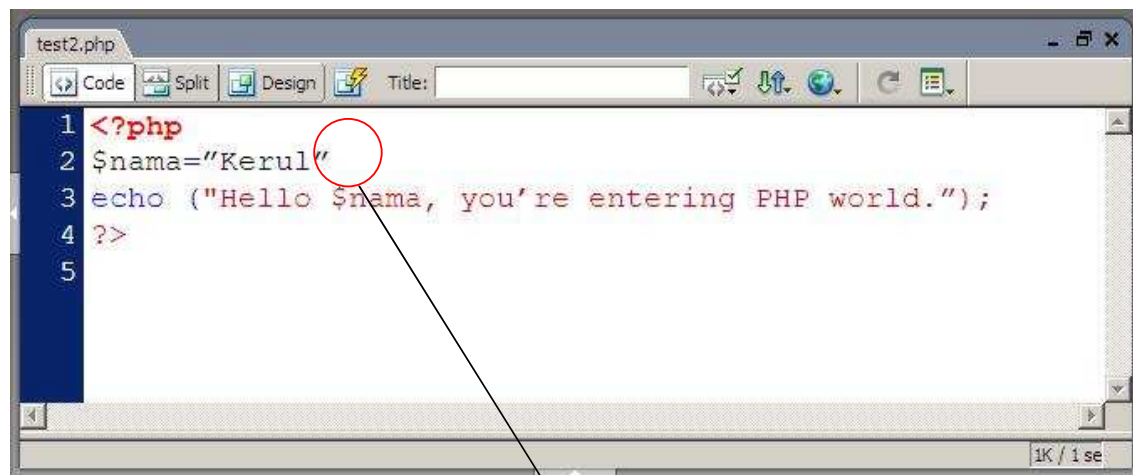
```
<html>
<head>
<title>Variable Testing </title>
</head>
<body>
This is the second test<br>
<?php
    $name = "ur name";
    $age=21;
    echo ("My name is $name <br>");
    echo ("I\'m $age years old...<br>");
?>
</body>
</html>
```

J.: SIMPLE DEBUGGING TECHNIQUE

If let say, you forgot to put the semicolon, there will be an error message. You need to find and correct the error, save the script file, and refresh the page in the browser.

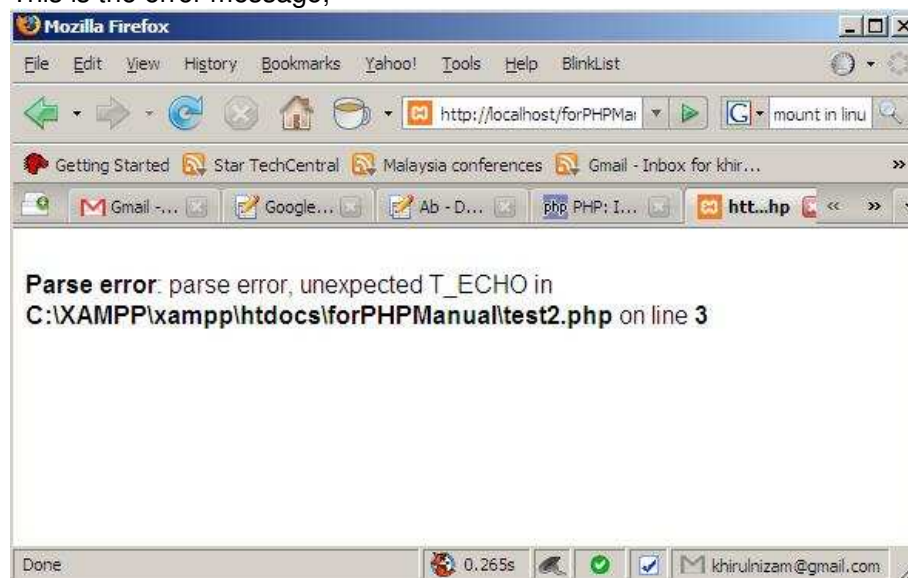
Example : this is the script with error.

```
<?php
    $nama="Kerul"
    echo ("Hello $nama, you're entering PHP world.");
?>
```



No semicolon, it is a syntax error

This is the error message;



Go to the script editor, and add a semicolon at the end of the variable initialization. Save the script and refresh the content in the browser. The same method goes to any error occur during the coding. Good luck, you need it very much.

Exercise 2

1. Decide whether the following variables are **valid** or **invalid**. Give your reason.

- a) \$apa
- b) \$123lekuk
- c) \$what-is-the-number
- d) \$_uji1
- e) \$nama saya
- f) \$For
- g) IF
- h) \$float
- i) \$variable_to_hold_names_or_matrix_for_students
- j) \$hasil*nombor

2. Decide whether the following initializations are **valid** or **invalid**. Give your reason.

- a) \$nom1 = 1231234;
- b) \$num3 = 23.7684;
- c) \$var1 = 123.0;
- d) \$dah_makan = TRUE;
- e) \$huruf = 't';
- f) \$alphabet = "B";
- g) \$pi = 0.0D;
- h) \$besar = 12345674567L;
- i) \$Lekuk123 = 12.5f;
- j) \$name = "Mamat";

Please refer the the declaration below to answer question 3 and 4.

```
$nom1 = 10;  
$nomPerpFLOAT = 10.0055;  
$nama_saya = "CRACKER";
```

3. Write the output for the following statements.

- a) echo ("Saya baru nak belajar PHP.");
- b) echo ("Hari ini saya bangun pukul " . \$nom1 . " pagi.");
- c) echo (" Where do you wanna go, ". \$nama_saya);
- d) echo ("Welcome to PHP \n world");

4. Write the *echo()* statement to display the following message. Use only one *echo()* function to display the message. Replace the value with the variable name from the above declaration.

- a) Saya suka nombor **10**.
- b) My nick name is **CRACKER**.
How about yours?
- c) Casting is needed to do this operation : **10/10.0055**.