PHP Variables

Variable is nothing it is just name of the memory location. A Variable is simply a container that is used to store both numeric and non-numeric information.

Rules for Variable declaration

1. Variables in PHP starts with a dollar($) sign, followed by the name of the variable.
2. The variable name must begin with a letter or the underscore character.
3. A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _)
4. A variable name should not contain space

Assigning Values to Variables

Assigning a value to a variable in PHP is quite east: use the equality(=) symbol, which also to the PHP's assignment operators.
This assign value on the right side of the equation to the variable on the left.
A variable is created the moment you assign a value to it:

Eg

```
<?php
  $myCar = "Honda";
  echo $myCar;
?>
```

Output- **Honda**

In the above example Create a variable **($mycar)** containing a string with **value="Honda".** To print the car name pass **$mycar** inside echo statement.

PHP Concatenation

Eg ii (concatenate variable with string)

```
<?php
  $myCar = "Honda City";
  echo $myCar . " is riding"; // dot(.) operator is used to concatenate mycar variable
and is riding
?>
```

Output- **Honda City is riding**

In the above example Variable($mycar) hold value="honda city". Now we wants to concatenate variable with string. pass this variable($mycar) inside echo statement. To concatenate this with a string("is riding") use dot(.) between variable name and string. The output will be displayed : Honda City is riding

Eg iii (Sum of two numbers)

```php
<?php
$first = 100;
$second = 200;
$third = $first + $second;
echo "Sum = ".$third;
?>
```

Output- **Sum = 300**

In the above example Declare $first , $second variable with value=100, 200 respectively. Now add these two numbers using arithmetic operator ("+"). sum of these two variable result stored in a third variable($sum). Now print the sum passing ($third) with echo statement with a string.

Eg iv (Subtraction of two numbers)

```php
<?php
$first = 1000;
$second = 500;
$third = $first - $second;
echo "Subtraction = ".$third;
?>
```

Output- **Subtraction = 500**

In the above example We perform subtraction using variables($first, $second) with vale=1000,500. Subtract second variable from first, result is hold by third variable($third) . Print this third variable passing with echo statement.

Destroying PHP Variables

To destroy a variable, pass the variable to PHP's unset( ) function. as in the following example:

Eg v

```
<?php
$name="steve";
echo $name;
//unset( ) function destroy the variable reference.
unset($name);
?>
```

Output steve

In the above example declare variable $name hold value="steve". In this program we used unset() function to delete a particular variable. first it show the output: "steve", because we pass unset function after echo statement. Now pass variable name inside unset($name) function output will show an Notice error(Variable is undefined).

Eg vi

```
<?php
$first = 100;
$second = 200;
$third = $first + $second;
echo "Sum = ".$third;
unset($third);
//after delete the variable call it again to test
echo "Sum = ".$third;
?>
```

Output Sum = 300
Sum = Notice error undefined third variable

Note : Trying to access or use a variable that's been unset( ), as in the preceding script, will result in a PHP "undefined variable" error message. This message may or may not be visible in the output page, depending on how your PHP error reporting level is configured.

Variable names in PHP are case-sensitive

```
<?php
$name="rexx";
$NAME="rahul";
echo $name."<br/>";
echo $NAME;
?>
```

Output rexx Rahul

Variable names in PHP are case-sensitive. As a result, $name refers to a different variable than does $NAME.

Inspecting Variable Contents(Variable Property)

PHP offers the var_**dump( )** function, which accepts a variable and X-rays it for you. Here's an example

Eg vii (For String value)

```
<?php
//define variables
$name = "Fiona";
$age=25;
//display variable contents
var_dump ($name);
var_dump($age);
?>
```

Output string 'Fiona' (length=5) int 25

In the above example We use var_dump( ) function to check the Contents(property) of variable, $name hold a string value ="Fiona" while $age hold an integer value = 25. Now pass this variable inside var_dump($name,$age) function . It show all information related this(data type of variable, length(only string), value)

Eg viii (For Integer values)

```
<?php
$first = 100;
$second = 200;
$third = $first + $second;
var_dump ($third);
?>
```

Output int 300

Eg ix (For Floating value)

```
<?php
$first = 100.5;
$second = 200.2;
$third = $first + $second;
var_dump ($third);
?>
```

Output float 300.7

In the above example variables first hold $first=100.5, second hold $second=200.2. Now add these two values, result is stored in third variable($third). Pass this variable inside var_dump($third) to check the content. output will float 300.7

Eg x (For Boolean value)

```
<?php
$bool = true;
var_dump ($bool);
?>
```

Output Boolean true

In the above variable $bool with value="true". var_dump($bool) function is used to display the result so output will display Boolean true, because variable holds a Boolean value