



The numbers used along with operators to form expressions are known as “operands” – in the expression `2 + 3` the numbers `2` and `3` are the operands.

Doing arithmetic

The arithmetic operators commonly used in PHP scripts are listed in the table below, together with the operation they perform:

Operator:	Operation:
<code>+</code>	Addition
<code>-</code>	Subtraction
<code>*</code>	Multiplication
<code>/</code>	Division
<code>%</code>	Modulus
<code>++</code>	Increment
<code>--</code>	Decrement

Operators for addition, subtraction, multiplication, and division act as you would expect. Care must be taken, however, to group expressions for clarity where more than one operator is used:

`a = b * c - d % e / f ;` **# This is unclear.**

`a = (b * c) - ((d % e) / f) ;` **# This is clearer.**

The `%` modulus operator divides the first given number by the second given number and returns the remainder of the operation. This is useful to determine if a number has an odd or even value. The `++` increment operator and `--` decrement operator alter the given number by one and return the resulting new value. These are most often used to count iterations in a loop and may be placed before the operand (“prefix”) or after the operand (“postfix”). The increment operator increases the value by one, and the decrement operator decreases the value by one. Shorthand expressions can usefully be created by combining an arithmetic operator with the `=` assignment operator. For example:

`a += b ;` **# Equivalent to `a = (a + b) ;`**

`a -= b ;` **# Equivalent to `a = (a - b) ;`**

`a *= b ;` **# Equivalent to `a = (a * b) ;`**

`a /= b ;` **# Equivalent to `a = (a / b) ;`**



Group expressions with parentheses to specify evaluation precedence – innermost groups get evaluated first.

...cont'd

- 1 Create a valid HTML document, like the one listed on page 16, then insert PHP tags into the body section

```
<?php  
# Statements to be inserted here.  
?>
```

- 2 Now, between the PHP tags insert statements to create and initialize two variables

```
$a = 4 ;  
$b = 8 ;
```

- 3 Next, insert statements to display the results of arithmetical operations using the variable values

```
$result = $a + $b ; echo "Addition : $result <br>" ;  
$result = $b - $a ; echo "Subtraction : $result <br>" ;  
$result = $a * $b ; echo "Multiplication : $result <br>" ;  
$result = $b / $a ; echo "Division : $result <br>" ;  
$result = $a % $b ; echo "Modulo : $result <br>" ;
```

- 4 Finally, insert statements to increment and decrement the values stored in the variables

```
$a++ ; echo "Increment : $a <br>" ;  
$b-- ; echo "Decrement : $b <br>" ;
```

- 5 Save the document in your web browser's `/htdocs` directory as `arithmetic.php`, then open the page via HTTP to see the results and variable values get displayed



```
PHP Arithmetic  
localhost/arithmetic.php  
Addition : 12  
Subtraction : 4  
Multiplication : 32  
Division : 2  
Modulo : 4  
Increment : 5  
Decrement : 7
```



arithmetic.php



PHP has some useful functions to work with numbers. For example, `round($n)` rounds `$n` to the nearest integer, and `number_format($n)` adds commas to `$n` denoting groups of thousands.



Notice that the variable value is immediately increased only with the prefix increment operator – with the postfix operator it is incremented the next time it is referenced.