

PERL MISCELLANEOUS OPERATORS EXAMPLE

http://www.tutorialspoint.com/perl/miscellaneous_operators_example.htm

Copyright © tutorialspoint.com

There are following miscellaneous operators supported by Perl language. Assume variable *a* holds 10 and variable *b* holds 20 then –

Operator	Description	Example
.	Binary operator dot . concatenates two strings.	If <i>a</i> = " abc " , <i>b</i> ="def" then <i>a. b</i> will give "abcdef"
x	The repetition operator <i>x</i> returns a string consisting of the left operand repeated the number of times specified by the right operand.	'-'x3 will give ---.
..	The range operator .. returns a list of values counting <i>upbyones</i> from the left value to the right value	2..5 will give 2, 3, 4, 5
++	Auto Increment operator increases integer value by one	<i>\$a++</i> will give 11
--	Auto Decrement operator decreases integer value by one	<i>\$a--</i> will give 9
->	The arrow operator is mostly used in dereferencing a method or variable from an object or a class name	<i>obj -> a</i> is an example to access variable <i>a</i> from object <i>obj</i> .

Example

Try the following example to understand all the miscellaneous operators available in Perl. Copy and paste the following Perl program in *test.pl* file and execute this program.

```
#!/usr/local/bin/perl

$a = "abc";
$b = "def";

print "Value of \$a = $a and value of \$b = $b\n";

$c = $a . $b;
print "Value of \$a . \$b = $c\n";

$c = "-" x 3;
print "Value of \"-\" x 3 = $c\n";

@c = (2..5);
print "Value of (2..5) = @c\n";

$a = 10;
$b = 15;
print "Value of \$a = $a and value of \$b = $b\n";

$a++;
$c = $a ;
print "Value of \$a after \$a++ = $c\n";

$b--;
$c = $b ;
print "Value of \$b after \$b-- = $c\n";
```

When the above code is executed, it produces the following result –

```
Value of $a = abc and value of $b = def  
Value of $a . $b = abcdef  
Value of "-" x 3 = ---  
Value of (2..5) = 2 3 4 5  
Value of $a = 10 and value of $b = 15  
Value of $a after $a++ = 11  
Value of $b after $b-- = 14
```

We will explain `-->` operator when we will discuss about Perl Object and Classes.

Processing math: 100%