Maths Function

- abs()

The abs() function returns the absolute value of a number.

abs(x)

Parameter	Description
x	Required. A number. If the number is of type float, the return type is also float, otherwise it is integer

Example

```
<?php
echo(abs(6.7) . "<br />");
echo(abs(-3) . "<br />");
echo(abs(3));
?>
```

The output of the code above will be:

```
6.7
3
3
```

- ceil()

The ceil() function returns the value of a number rounded UPWARDS to the nearest integer.

ceil(x)

Example

In the following example we will use the ceil() function on different numbers:

```
<?php
echo(ceil(0.60) . "<br />");
echo(ceil(0.40) . "<br />");
echo(ceil(5.1) . "<br />");
echo(ceil(-5.1) . "<br />");
echo(ceil(-5.9))
?>
```

The output of the code above will be:

```
1
1
6
-5
-5
```

- ## floor()

The floor() function returns the value of a number rounded DOWNWARDS to the nearest integer.

floor(x)

Example

```
<?php
echo(floor(0.60) . "<br />");
echo(floor(0.40) . "<br />");
echo(floor(5) . "<br />");
echo(floor(5.1) . "<br />");
echo(floor(-5.1) . "<br />");
echo(floor(-5.9))
?>
```

The output of the code above will be:

```
0
0
5
5
-6
-6
```

- ## fmod()

The fmod() function divides x by y and returns the remainder (modulo) of the division.

fmod(x,y)

Parameter	Description
x	Required. A number
y	Required. A number

Example

In the following example we will use the fmod() function to return the remainder of 5/2:

```
<?php
$r = fmod(5,2);
echo $r
?>
```

The output of the code above will be:

```
1
```

- ## max()

The max() function returns the number with the highest value of two specified numbers.

max(x,y)

Example

```
<?php
echo(max(5,7) . "<br />");
echo(max(-3,5) . "<br />");
echo(max(-3,-5) . "<br />");
echo(max(7.25,7.30))
?>
```

The output of the code above will be:

```
7
5
-3
7.3
```

- min()

The min() function returns the number with the lowest value of two specified numbers.

```
min(x,y)
```

Example

```
<?php
echo(min(5,7) . "<br />");
echo(min(-3,5) . "<br />");
echo(min(-3,-5) . "<br />");
echo(min(7.25,7.30))
?>
```

The output of the code above will be:

```
5
-3
-5
7.25
```

- pow()

The pow() function raises the first argument to the power of the second argument, and returns the result.

```
pow(x,y)
```

Note: The pow() function will return -1.#IND if the parameter y is a floating number.

Example

```
<?php
echo pow(4,2) . "<br />";
echo pow(6,2) . "<br />";
echo pow(-6,2) . "<br />";
echo pow(-6,-2) . "<br />";
echo pow(-6,5.5);
?>
```

The output of the code above will be:

```
16
36
36
0.0277777777778
-1.#IND
```

- ## sqrt()

The sqrt() function returns the square root of a number.

sqrt(x)

Note: The sqrt() function will return -1.#IND if the parameter x is a negative number.

Example

```
<?php
echo(sqrt(0) . "<br />");
echo(sqrt(1) . "<br />");
echo(sqrt(9) . "<br />");
echo(sqrt(0.64) . "<br />");
echo(sqrt(-9))
?>
```

The output of the code above will be:

```
0
1
3
0.8
-1.#IND
```

- ## rand()

The rand() function generates a random integer. If this function is called without parameters, it returns a random integer between 0 and RAND_MAX. If you want a random number between 10 and 100 (inclusive), use rand (10,100).

rand(min,max)

Parameter	Description
min,max	Optional. Specifies the range the random number should lie within

Note: On some platforms (such as Windows) RAND_MAX is only 32768. So, if you require a range larger than 32768, you can specify min and max.

Example

```
<?php
echo(rand() . "<br />");
echo(rand() . "<br />");
echo(rand(10,100))
?>
```

The output of the code above could be:

```
17757
3794
97
```

- round()

The round() function rounds a number to the nearest integer.

round(x,prec)

Parameter	Description
x	Required. The number to be round
prec	Optional. The number of digits after the decimal point

Example

```
<?php
echo(round(0.60) . "<br />");
echo(round(0.50) . "<br />");
echo(round(0.49) . "<br />");
echo(round(-4.40) . "<br />");
echo(round(-4.60))
?>
```

The output of the code above will be:

```
1
1
0
-4
-5
```

- pi()

The pi() function returns the value of PI.

pi()

Example

```
<?php
echo pi();
?>
```

The output of the code above will be:

```
3.14159265359
```