

is_numeric

WMPRO, WMMINI FW >= 1.0 WMMEGA FW >= 2.0

Check if a value is numeric (**int**, **float** or numeric **string**)

Description

```
int is_numeric ( mixed $value )
```

Check to see if a value is numeric

Parameter

\$value: Variable or expression to be evaluated

Return Value

1 (true) if numeric, 0 (false) if not numeric (arrays always return 0)

Examples

```
<?
 $f=1.0;
 print(is_numeric($f)); // outputs 1, $f is a float
 $i=1;
 print(is_numeric($i)); // outputs 1, $i is an int
 $nns="cool";
 print(is_numeric($nns)); // outputs 0, $nns is a non-numeric string
 $ns="-1.5";
 print(is_numeric($ns)); // outputs 1, $ns is a numeric string
?>
```

Set up an array of different types of values. Then iterate through the array and output each element type, value and whether is_numeric() returns true or false:

```
<pre><?
 $tests = array(
     1337,
     0x539, // 1337 expressed in hexadecimal format
     02471, // 1337 expressed in octal format
     0b10100111001, // 1337 expressed in binary format
     -25,
 );
```

```
42.8,  
-365.0,  
"42.9",  
"no numbers",  
"123 with numbers",  
array(1,2,3),  
);  
for ($i=0;$i<sizeof($tests);$i++) {  
    if (is_int($tests[$i])) print("The int");  
    if (is_float($tests[$i])) print("The float");  
    if (is_string($tests[$i])) print("The string");  
    if (is_array($tests[$i])) print("The array");  
    if (!is_array($tests[$i])) print(" '$tests[$i] '");  
    if (is_numeric($tests[$i])) {  
        print(" is numeric\r\n");  
    } else {  
        print(" is NOT numeric\r\n");  
    }  
}  
?></pre>
```

The above example will output:

```
The int '1337' is numeric  
The int '1337' is numeric  
The int '1337' is numeric  
The int '1337' is numeric  
The int '-25' is numeric  
The float '42.799999' is numeric  
The float '-365.011138' is numeric  
The string '42.9' is numeric  
The string 'no numbers' is NOT numeric  
The string '123 with numbers' is NOT numeric  
The array is NOT numeric
```

Notes

Unlike functions [is_int\(\)](#), [is_float\(\)](#), [is_string\(\)](#) and [is_array\(\)](#) this function looks at more than just the type of the variable, it also checks strings to see if they are numeric.

See Also

[is_int\(\)](#) - Check if a variable is an **integer**

[is_float\(\)](#) - Check if a variable is a **float**

[is_string\(\)](#) - Check if a variable is a **string**

[is_array\(\)](#) - Check if a variable is an **array**

[isset\(\)](#) - Check if a variable exists

[uPHP Numbers and Numeric Calculations](#)

[uPHP Variable Types and Limits](#)

[array\(\)](#) - Create an **array**, with optional values

From:

<http://www.wattmon.com/dokuwiki/> - **Wattmon Documentation Wiki**

Permanent link:

http://www.wattmon.com/dokuwiki/uphp/functions/is_numeric?rev=1491832304

Last update: **2021/09/13 05:56**

