

# explode

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

Turn a **string** into an **array**

## Description

**array** explode ( **string** \$value, **string** \$delimiter )

Turn a **string** into an **array** by separating out the elements as bounded by a delimiter character

## Parameters

**\$value**: **String** to be separated into elements in the **array**

**\$delimiter**: Character to use for separation

## Return Values

**Array** of elements

## Example

```
<pre><?
    $mylist="one,two,three,four";
    $myarray=explode($mylist,",");
    print_r($myarray);
?></pre>
```

The above example will output:

```
[$myarray] = Array (
    (string) [0] => one
    (string) [1] => two
    (string) [2] => three
    (string) [3] => four
)
?></pre>
```

Note that if only one element is present (without the separator character) then a value of -1 will be returned. This differs from the PHP implementation.

```
<pre><?
    $mylist="one";
```

```
$myarray=explode($mylist,",");

// the fix for a single parameter is to use is_array
if (!is_array($myarray)) {
    $myarray=array($mylist);
}
print_r($myarray)
```

## See Also

[implode\(\)](#) - Turn an **array** into a **string**

[print\\_r\(\)](#) - Dump the contents of an **array** to the current output

[sizeof\(\)](#) - Return the number of elements in an **array**

[uPHP Variable Types and Limits](#)

## Additional Information

Note that there is an important difference in the parameter order between mainline PHP and Wattmon uPHP. In the mainline PHP function the delimiter comes first, but in uPHP the **string** to be exploded comes first:

PHP `explode(delimiter, string)`

uPHP `explode(string, delimiter)`

Wattmon uPHP's parameter order is easier to remember and more consistent: The delimiter parameter comes last in both [explode\(\)](#) and [implode\(\)](#).

From:

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

Permanent link:

<http://wattmon.com/dokuwiki/uphp/functions/explode?rev=1612260868>

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

